PERFORMANCE ANALYSIS OF PROTOCOL INDEPENDENT MULTICASTING-DENSE MODE IN LOW EARTH ORBIT SATELLITE NETWORKS

THESIS

Mark D. Saeger, First Lieutenant, USAF

AFIT/GCE/ENG/03-03

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

AFIT/GCE/ENG/03-03

PERFORMANCE ANALYSIS OF PROTOCOL INDEPENDENT MULTICASTING-DENSE MODE IN LOW EARTH ORBIT SATELLITE NETWORKS

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Computer Engineering

Mark D. Saeger, BS

First Lieutenant, USAF

March 2003

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AFIT/GCE/ENG/03-03

PERFORMANCE ANALYSIS OF PROTOCOL INDEPENDENT MULTICASTING-DENSE MODE IN LOW EARTH ORBIT SATELLITE NETWORKS

Mark D. Saeger, BS
First Lieutenant, USAF

Approved:

Dr. Richard A. Raines (Chairman)	12 Mar 03 date
Major Rusty O. Baldwin, PhD (Member)	13 Mar 03 date
Dr. Michael A. Temple (Member)	12 Mar 03 date

Acknowledgments

I would like to express my sincere appreciation to my faculty advisor, Dr. Richard Raines, for his guidance and support throughout the course of this thesis effort. The insights and experience he provided were invaluable. I would also like to thank my committee members, Major Rusty Baldwin and Dr. Michael Temple for their guidance and knowledge.

Last, but certainly not least, I would like to thank my lovely wife for her never-ending support throughout this whole thesis process. She supported me every day by taking care of all of the little things I no longer had time to focus on and always had confidence and trust that I would finish this thesis effort.

Mark D. Saeger

Table of Contents

Page

List of Figures

Figure Page

List of Tables

AFIT/GCE/ENG/03-03

Abstract

This research explored the implementation of Protocol Independent Multicasting – Dense Mode (PIM-DM) in a LEO satellite constellation. PIM-DM is a terrestrial protocol for distributing traffic efficiently between subscriber nodes by combining data streams into a tree-based structure, spreading from the root of the tree to the branches. Using this structure, a minimum number of connections are required to transfer data, decreasing the load on intermediate satellite routers.

The PIM-DM protocol was developed for terrestrial systems and this research implemented an adaptation of this protocol in a satellite system. This research examined the PIM-DM performance characteristics which were compared to earlier work for On-Demand Multicast Routing Protocol (ODMRP) and Distance Vector Multicasting Routing Protocol (DVMRP) – all in a LEO satellite network environment.

Experimental results show that PIM-DM is extremely scalable and has equivalent performance across diverse workloads. Three performance metrics are used to determine protocol performance in the dynamic LEO satellite environment, including Data-to-Overhead ratio, Received-to-Sent ratio, and End-to-End Delay. The OPNET® simulations show that the PIM-DM Data-to-Overhead ratio is approximately 80% and the protocol reliability is extremely high, achieving a Receive-to-Sent ratio of 99.98% across all loading levels. Finally, the PIM-DM protocol introduces minimal delay, exhibiting an average End-to-End Delay of approximately 76 ms; this is well within the time necessary to support real-time communications. Though fundamental differences between the DVMRP, ODMRP, and PIM-DM implementations precluded a direct comparison for each experiment, by comparing average values, PIM-DM generally provides equivalent or better performance.

PERFORMANCE ANALYSIS OF PROTOCOL INDEPENDENT MULTICASTING-DENSE MODE IN LOW EARTH ORBIT SATELLITE NETWORKS

1. Introduction

Networks, and more specifically the Internet, dominate many aspects of our daily life. The ability to rapidly "surf the net" for an esoteric nugget of information, web page, video clip, or sound bite has become the rule for many, rather than the exception. This global capability highlights the need for efficient means of transferring data over large distances, especially if identical information is flowing to adjacent or closely situated neighbors.

The advent of satellites capable of serving as routers (just as there are terrestrial routers in a network) adds great flexibility to the communications infrastructure. The network is no longer restricted by wires or line-of-sight distances, but instead can span the entire globe.

The cost for a satellite network far exceeds the cost of a terrestrial network, so efficiently utilizing the network is critical. Speed and large transmission capabilities are important features, but if the network is not used extremely efficiently, cost and wasted resources are a potential problem. Leveraging the benefits of satellite networks requires using some protocol enhancements developed to efficiently transmit data on terrestrial networks.

1.1 Background

In a traditional network environment, if 100 users located in roughly the same geographic region want to view an identical video clip, 100 point-to-point unicast connections from the server to each of the users are established. These 100 connections

may follow the same communication links (i.e., routers) most of the way to the destination. The information gets to the users, but with a significant amount of wasted bandwidth as the routers pass the same packet 100 times. Multicasting organizes these 100 users into a tree structure, and transfers the information as single connections until a branch in the tree is reached. At a branch, the data stream is divided into two (or more) streams before continuing to the next branch, causing another division. Network efficiency is realized by combining multiple data streams for as long as possible when there is a common route and common data stream.

The concept of multicasting [Dee89] has grown to encompass a wide range of protocols, each with its own strengths and weaknesses. The implementation may differ between protocols, but each has a goal of efficiently transferring a stream of information to multiple users by creating a minimum number of unique connections.

This idea can be further extended to users scattered over a wide geographic region. Through careful subscription management, the routers can combine multiple data streams into a single data stream for efficient network communication between these diverse user bases.

1.2 Research Problem

Efficiently transmitting information is a widely researched area for terrestrial networks. Satellite networks have not received the same attention and therefore there is little research in this area. There are many commonalities between a terrestrial and a satellite communications network. Many protocol enhancements and even the protocol itself can be applied across both networks, although some changes may be required to

account for idiosyncrasies of a specific network. Ultimately, the intent is to efficiently transfer information through both networks and across network boundaries.

There are a number of multicasting protocols that are implemented or proposed for satellite networks. Prior research investigated both ODMPR and DVMRP [Tho01]. This investigation extends the previous work by modeling and analyzing the performance of Protocol Independent Multicasting-Dense Mode (PIM-DM) [DEF99, ANS02].

The application of packet transmissions over a satellite network, and specifically applying PIM-DM to the network, has not been widely studied. Meshing the ideas for PIM-DM taken from a terrestrial setting and incorporating the same basic protocol and behavior is the focus of this research. PIM-DM is simulated in a low earth orbit (LEO) satellite network constellation to observe the protocol behavior, see how efficiently the network transmits information, and determine how much overhead is required to maintain the network.

1.3 Scope

The scope of this thesis extends to examining the network behavior of PIM-DM in a LEO satellite environment. PIM-DM network performance is defined by the following three statistics: data-to-overhead ratio, receive-to-sent ratio, and end-to-end delay. To support PIM-DM, a unicast network model is developed for the underlying transmission structure and the following ratio statistics are gathered: data-to-overhead and receive-to-sent. An underlying unicast protocol must be present to support PIM, but the particular unicast protocol is not tied directly to PIM.

Network behavior is modeled using various workloads, membership levels, and geographic locations. The satellite network being modeled is based on the Iridium® satellite constellation.

Lastly, the network behavior is simulated using a discrete time network event simulation tool. The system being explored has not been built and this research focuses on how a system based on PIM-DM would behave if it were fielded in a LEO satellite environment.

1.4 Approach

This thesis extends the work of Pratt [Pra99], Thomas [Tho01], and Fossa [Fos01]. Pratt and Fossa explored unicast routing in a LEO satellite configuration, whereas Fossa examined error conditions and Pratt extended the model to include dynamic routing. Thomas expands this work by implementing the multicasting protocols DVMRP and ODMRP in a LEO environment.

The objective of this thesis is to add PIM-DM to the network model. The network as modeled by Thomas [Tho01] is taken as the base and modified to incorporate PIM. Additionally, to support PIM-DM, an underlying unicast model is implemented.

The network is based on the Iridium® satellite constellation with ground nodes placed on all continents. The constellation is modeled using a discrete time network event simulation tool. The simulation models the physical satellite network and the network traffic generated by the multicasting algorithm.

1.5 Summary

The remainder of this document is organized into four chapters. Chapter 2 contains the literature review where background associated with multicasting is presented. The methodology for the experimental phase of this investigation is given in Chapter 3. The analysis of the results and comparison to earlier works follow in Chapter 4. Finally, Chapter 5 provided a summary of the thesis effort and identifies areas of the research to be explored in future research efforts.

2. Literature Review

2.1 Introduction

This chapter provides a tiered overview of pertinent literature relating to satellite communication and more specifically, multicast communications over satellite networks. This chapter is organized into four areas, starting with broad-based communications in mobile and satellite networks, followed by a discussion of lower level protocols and communication standards which facilitate the transfer of information via a satellite network. Expanding on this background information, an overview of four common methods of network communication is provided. Finally, this chapter closes with an in-depth review of various aspects of multicast communications.

Network communications play an integral role in all aspects of daily life, ranging from home use, business, military, and the government. Advances in technology have made the Internet-in-the-Sky a reality and introduced problems not encountered in terrestrial networks. Traditional terrestrial networks are typically built on a static communications infrastructure. That is, the overall network topology may change, but the underlying node structure remains relatively stable. Each network node is connected via either fiber optic or copper cabling, with long-range connections possibly using microwave or satellite links. Ultimately, communications are terminated in a relatively stationary location.

Mobile and satellite communications introduce a plethora of problems which do not affect traditional terrestrial networks. The most obvious of these is mobility, since both mobile and satellite networks have nodes that are constantly in motion. The mobility issue is even more profound in satellite networks – nodes could be moving at 7.5

km/s, much faster than any terrestrial mobile communications link. Even with the greater difference in node speeds, mobile and satellite communications are closely related in terms of design issues and implementation challenges. Many of the issues are inversely related to each other. For example, in mobile communication, client nodes are mobile while host nodes tend to be stationary. Wireless mobile communications are typically considered to be ad-hoc - there is no fixed infrastructure. Node and network administration is dynamic and reconfigurable [BLS00]. Satellite communications, on the other hand, have stationary clients at the ground stations or in geosynchronous orbit, with Low Earth Orbit (LEO) and Medium Earth Orbit (MEO) host nodes moving rapidly around the earth. Even with this dynamic network topology, the satellite configuration is predictable due to the constant satellite orbits. This greatly simplifies satellite tracking and identification at any moment in time [RaS96]. Satellite mobility makes design issues more complicated while providing an irreplaceable communications service. A constellation of satellites can provide communications facilities for diverse areas of the world that do not have the resources, permission, or infrastructure to lay out ground based networks. Additionally, satellites provide global reach for critical communications since satellites operate in an area not considered to belong to any nation.

2.2 Satellite Systems Overview

Modern satellite communications has various orbital altitudes and constellation configurations. Four main orbital configurations comprise a large percentage of systems currently used. These configurations are: Geostationary (GEO), Medium Earth Orbit (MEO), Highly Elliptical Orbit (HEO), and Low Earth Orbit (LEO). The mechanics of satellite motion, orbital specific delays, and an overview of coverage areas sets the stage

for understanding the nature of how satellites can be utilized to provide networking services.

2.3 Orbital Synchronicity

Satellites orbits fall into two broad categories [Tom98] – non-synchronous or geosynchronous. A non-synchronous satellite orbit is in constant motion and is never stationary relative to a reference location on earth. A geosynchronous satellite is moving at the same angular velocity as the earth and stays in a specific location relative to a reference location on earth. Tracking and communication issues related to geosynchronous satellites are less complex than those pertaining to non-synchronous orbits. For example, non-synchronous satellites must be tracked when they are visible to a ground station and any communications must take place within this window of visibility. A geosynchronous satellite always appears in the same location, so any ground station in the satellite's field-of-view can view the satellite and always communicate with it.

Satellite velocity v_s can be determined from Equation 2.1 using the gravitational constant $G = 6.6720\times10^{-11}\, m^3/kg\cdot s^2$, the earth's mass $M_E = 5.9742\times10^{24}\, kg$, and the satellite's orbital altitude a_s (measured from the center of the earth), and the radius of the earth $a_E = 6378km$.

$$v_S = \sqrt{\frac{GM_E}{a_S + a_E}} \qquad 2.1$$

Satellite period can be calculated from Equation 2.2 using the geocentric gravitational constant $\mu = 3.986005\times10^{14}\, m^3/\sec^2$.

$$P = \frac{2\pi}{\sqrt{\mu / (a_S + a_E)^3}} \qquad 2.2$$

Round trip communications time (RTT) is shown by Equation 2.3 where the speed-of-light in a vacuum is $c_{vacuum} = 2.9979 \times 10^8 \ m/s$.

$$RTT = \frac{2 \times a_s}{c_{vacuum}} \qquad 2.3$$

2.3.1 Non-synchronous Orbits

Non-synchronous satellites have three main orbital types: Low Earth Orbit (LEO), Medium Earth Orbit (MEO), and Highly Elliptical Orbit (HEO). Each orbit provides a unique area of coverage for the earth that is directly related to the satellite altitude. The closer a satellite is to the earth's surface, the smaller the coverage area from and the higher the number of satellites required to provide full earth coverage.

2.3.1.1 Low Earth Orbit (LEO)

LEO satellite altitudes range from 200 to 1000 km above sea level or 6,578 to 7,378 km from the center of the earth [Eva99]. Based on this data and using Equation 2.2, the LEO satellite period is between 1:28 and 1:44 hours. At this altitude and using Equation 2.1, the LEO satellite velocity is between 7.350 and 7.784 km/s. Due to the low orbital altitude of the LEO satellite and using Equation 2.3, there is a round trip propagation time of approximately 1.334 to 6.671 ms. The high velocity and small period means a LEO satellite is only visible for small periods of time as it passes within the field-of-view of a given earth station - usually 15 minutes or less. Various satellite

configurations provide whole earth coverage, ranging in constellation size of 66 for Iridium to 288 for Teledesic [CCS01].

2.3.1.2 Medium Earth Orbit (MEO)

MEO satellites usually have an altitude of about 10,000 km. Using Equations 2.1, 2.2, and 2.3, the MEO satellite period is approximately 6 hours with an orbital velocity of 4.933 km/s and a propagation delay of approximately 67 ms. MEO satellites provide a convenient middle area between GEO and LEO satellites. They have a greater coverage area (similar to GEO) with a smaller propagation delay (similar to LEO).

2.3.1.3 Highly Elliptical Orbit (HEO)

A highly elliptical orbit satellite traverses a path that brings the satellite relatively close to earth and then swings deep into space before making another pass close to the earth. There is an area of about 8,000 km between the LEO and MEO satellite orbital ranges containing the Van Allen radiation belt [Elb87]. This radiation belt does not permit long-term satellite operation due to the extra shielding required to protect sensitive satellite electrical components. Due to the extreme radiation conditions, satellites are not normally positioned in this belt for long periods of time, instead HEO satellites, for example, may move through this belt during their orbit. The most common example of this type of orbit is the Molnya orbit, which has an apogee of 40,000 km and a perigee of about 1,000 km [Tom98]. This type of satellite orbit is rather unusual and very few commercial applications exist which take advantage of this orbital pattern.

2.3.2 Geosynchronous Orbit (GEO)

The geosynchronous orbit is the only orbit that places the satellite over a fixed point on the earth's surface [Tom98]. This fixed point allows constant communication with no need to perform a communications handoff. Unlike lower earth orbit satellites, a single GEO satellite can provide coverage for a large portion of the earth. The period of a geosynchronous satellite is 23 hours, 56 minutes, 4 seconds mean solar time (P=86,164s) and the geocentric gravitational constant $\mu = 3.986005 \times 10^{14}\ m^3/\sec^2$ [Rod89]. Inserting both of these numbers into Equation 2.4, yields the altitude of a geosynchronous satellite as referenced from the earth's surface a_s

$$a_S = \left(\frac{\mu P^2}{4\pi^2}\right)^{\frac{1}{3}} \qquad 2.4$$

$$a_{GEO} = a_S - a_E = 42{,}164 - 6{,}378 = 35{,}786 km \qquad 2.5$$

GEO satellites are placed into orbit around the equator of the earth at an altitude of approximately 35,786 km. Unlike the other orbits, a geostationary orbit is a limited resource - there is only one orbital altitude which will maintain this stationary position. Using Equations 2.1 and 2.3, the GEO satellite is moving at a velocity of 3.075 km/s with a roundtrip propagation delay of 0.2386 s. Of all satellite orbits in use, the GEO satellite has the highest propagation delay and is the least suitable for high-speed network communications. Almost whole earth coverage is possible using only three GEO satellites spaced 120 degrees apart. Except for locations near the poles, this configuration of satellites can provide communications between any two points on earth [Eva99].

2.4 Low Earth Orbit (LEO) Satellites

The next section of this chapter narrows the focus to LEO satellite networks and how space-based networking protocols use LEO networks. LEO satellites provide a capable platform for high-speed network communications but present unique challenges for designers.

2.4.1 Overview

Of all of the satellite constellations mentioned, LEO constellations have the lowest propagation delay between earth stations and satellites. The constellation's close proximity to the earth results in a propagation delay of approximately 1.334 to 6.671 ms roundtrip time. A roundtrip is defined as a communication event being transferred from the earth station to the satellite back to the earth station. Communication is also possible via Inter-Satellite Links (ISL), so the information may actually traverse a number of satellites before being transferred back to an earth station.

2.4.2 Transmission Speeds

Satellite networks provide transmission speeds ranging from 9600 bps to almost 622 Mbps (OC-12) while terrestrial networks currently support 2.4 Gbps (OC-48). Even with speeds slower than their terrestrial counterparts, the ability of a satellite network to transmit data across vast expanses of the earth with no physical wiring is appealing. Not only can the satellite transmit data to an arbitrary location on earth, the satellite can also broadcast that data over a large coverage area, potentially distributing the data to multiple ground stations or receivers.

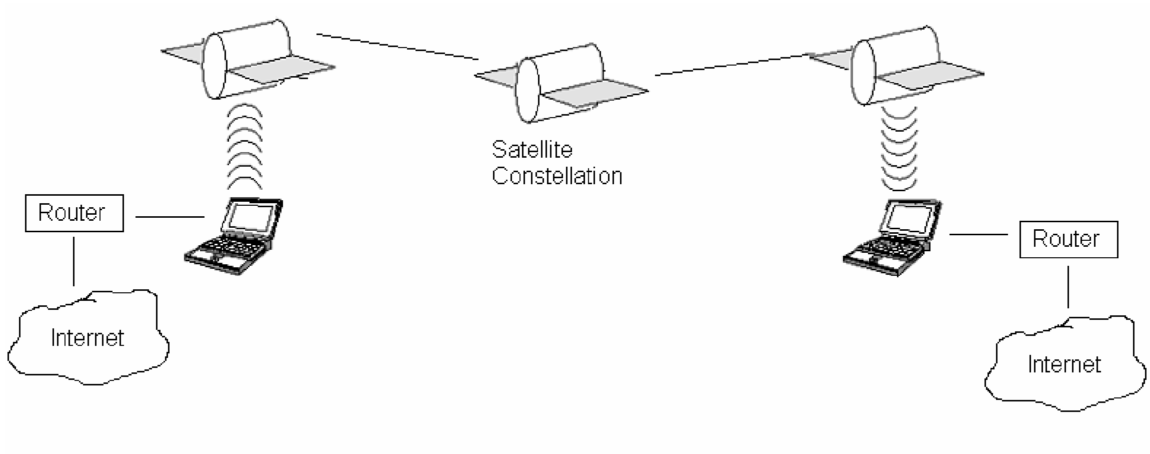
2.4.3 Transmission Control Protocol/Internet Protocol (TCP/IP)

The Transmission Control Protocol / Internet Protocol (TCP/IP) is the most widely used protocol in terrestrial networks. The TCP/IP protocol provides a reliable transport mechanism that ensures data delivery and adjusts its packet transmission rate based on the amount of bandwidth available for the connection. TCP/IP adjusts the packet transmission speed rate based on successful acknowledgements received by the sender from the receiver [Met99]. The number of packets transmitted before acknowledgement increases for every error-free transmission, and only after an error, does TCP scale back the packet transmission rate.

2.4.3.1 TCP/IP Transmission Overview

Successful satellite transmission of information using TCP/IP is described below and is illustrated in Figure 2-1. Information in the form electrical or light signals moves across a terrestrial network using either copper or fiber optic cabling. If the information will traverse a satellite link, a router at the earth station accepts the packet stream and turns it into a microwave signal that can be transmitted to the satellite. Once the earth station has locked onto the receiving satellite, the information is broadcast via a satellite dish to the satellite receiver. The satellite accepts the incoming packet stream, performs any necessary processing and amplifies the signal before forwarding the data to either another satellite or to a different earth station in the satellite's field of view. The information is received by the destination earth station and converted back to electrical or light impulses that flow back into the terrestrial network.

Figure 2-1 Ground to Satellite Network to Ground Communication



2.4.3.2 TCP/IP Factors Affecting Performance

TCP/IP over satellite has four main factors that determine the effectiveness of the TCP/IP protocol [CCS01]. These factors are not necessarily applicable to terrestrial TCP/IP networks since the original TCP/IP protocol did not account for the larger distances and lossy transmissions of a satellite network.

2.4.3.2.1 High Latency

A packet transmitted via a GEO satellite has a roundtrip time of 0.2386 seconds while one transmitted over a LEO satellite has a roundtrip time of between 1.334 and 6.671 ms. The latency of the satellite transmission affects TCP performance. TCP uses a slow start algorithm which gradually fills the network to determine a maximum window size, rather than trying to send an arbitrarily large amount of data which may cause network congestion and numerous retransmissions [ANV99]. The slow start algorithm increases the amount of data sent exponentially between successive acknowledgements until either an error occurs or the maximum window size is reached. One packet is transmitted and the sender must wait for an acknowledgement, then two packets are

transmitted before waiting for an acknowledgement, then four packets, and so on until an error occurs. Once an error occurs, the algorithm enters a congestion avoidance phase and attempts to add a single packet [ANV99]. The slow start algorithm can be expected to take RTT * Log (2 * Window size) to get out of slow start and into regular transmission [Met99]. In the case of a high latency communications link, the time between the sender transmitting the information and the sender receiving the acknowledgement causes inefficient packet transmission rates for TCP. The window size is the maximum number of packets that can be transmitted without an acknowledgement. Due to high latency, the slow start algorithm may take an excessive amount of time to fill the window causing inefficient communications.

2.4.3.2.2 Large bandwidth-delay product

The bandwidth multiplied by the latency (BW * RTT) determines how much data can be in-flight without receiving an acknowledgement [Met99]. Here again, the higher RTT determines how large this bandwidth delay product is, and how quickly TCP/IP can get out of slow start and use the link to its full capacity. Additionally, with a bigger bandwidth-delay product, buffer sizes for both the sender and receiver must be able to accommodate a large amount of unacknowledged data.

2.4.3.2.3 High bit-error rate (BER)

In terrestrial networks with a BER of approximately 10^{-14} [PeD00], a lost data packet is generally the result of congestion rather than error, causing TCP to enter congestion avoidance. In a satellite network, however, with BER rates of approximately 10^{-6} for unencoded transmissions [GhD99], a corrupted packet causes TCP to enter

congestion avoidance, decreasing the size of the transmit window unnecessarily. The high BER rates of a satellite network amplify the bursty nature of many satellite errors since TCP's correction mechanisms, such as fast retransmit, respond to such errors with a reduction in window size [GhD99]. Once TCP identifies what is believed to be congestion, it enters a slow start phase, throttling overall throughput as it adjusts to the new network load.

2.4.3.2.4 Variable roundtrip time (RTT)

There is no general measure of RTT in LEO satellite networks since LEO satellites rapidly move in and out of the ground stations visible window. Additionally, there is a handoff between satellites that must occur as satellites move out of a ground stations field of view. Finally, differences in satellite positions for inter-satellite communications affects RTT. All of these factors contribute to the variable RTT that in turn affects both the latency and bandwidth-delay product [CCS01].

2.4.3.3 TCP/IP Enhancements for Satellite Communications

There are a number of methods proposed to remedy or at least offset problems associated with TCP/IP and satellite networks. To facilitate efficient error recovery and avoid entering slow start, a fast retransmit algorithm should be implemented as stated in RFC2581 [ANV99]. Fast retransmit is invoked after the third successive acknowledgement for a missed packet, after which the sender retransmits the missing packet. After the retransmitted packet is received, a new acknowledgement is sent with the latest packet number that has been received, which effectively catches the acknowledgement window up to the current packet on the sender side. By following this

fast retransmit algorithm, only the missed packet is resent and subsequent packets that were received after the missed packet do not have to be resent. A larger initial window decreases the slow start time, but has no effect on the congestion avoidance phase [CCS01]. Implementing forward error correction (FEC) also helps TCP/IP over satellites. With FEC, the BER can be lowered to between 10^{-7} and 10^{-9} [GhD99]. Offsetting the higher BER of the satellite connection by correcting an error, a packet does not need to be retransmitted thus avoiding the possibility that TCP enters into slow start. Avoiding IP Fragmentation also improves the data to overhead ratio of a packet stream [Met99]. By using path maximum transfer unit (MTU) discovery, packet sizes can be sent that do not exceed this limit so fragmentation does not occur. The data to overhead ratio is greater when packets are not fragmented and this also avoids reassembling the fragmented packets, increasing throughput and performance. Selective Acknowledgement (SACK) is another means of efficiently notifying the sender about which packets have been received. With SACK, the sender is told which packets have been received correctly and, by their omission from the list, which packets need to be retransmitted. Thus SACK decreases the overhead due to multiple retransmission requests and more efficiently identifies the missing packets to the sender.

2.5 Routing mechanisms

TCP/IP forms the basis for the LEO satellite network just described. However, to move information between the satellite nodes, a more general routing mechanism must be in place. There are four different types of routing mechanisms, each suitable for certain types of traffic, and some which can be used to simulate or facilitate other routing mechanisms and the distribution of traffic. The four types are unicast, broadcast, anycast,

and multicast. Support for these four routing mechanisms differs between IPv4 and IPv6. IPv6 has robust support for unicast, anycast, and multicast (which replaced broadcast support) built-in, whereas IPv4 does not natively support these four methods, supporting only unicast and broadcast.

2.5.1 Unicast

The most common means of communication takes place between a single sender and a single receiver. This communications is termed unicast. Unicast has two specific endpoints; one endpoint is the sender, the other is the receiver. The traffic flows as a stream between this single sender and the single receiver. This is a logical stream, and as such, packets may take different routes due to congestion or link failure, but ultimately the traffic is confined to these two specific communications endpoints. If there are only two nodes that need to communicate and both are aware of the other, this may be the most efficient method of transferring information. Traffic flows between the two interested nodes and no other node receives the information. A unicast connection can span a small local subnet on a Local Area Network (LAN) to a connection spanning multiple subnets and domains on a Wide Area Network (WAN). Unicast transmission streams handle much of the communication traffic on the Internet. Under IPv4, Table 2-1 shows the dotted-decimal IP addresses illustrating address classes. IPv6 uses a 128-bit address field, where a high octet of FF denotes multicast and all other address ranges indicate unicast. Under IPv6, there is no specific range for anycast, instead anycast addresses are a subset of the unicast address space [Tay98].

Table 2-1 IPv4 Address Format

IP Class of Addresses	Address Format	Address Range
Class A	[0][NetID(7)][HostID (24)]	0.0.0.0 to 7F.FF.FF.FF
Class B	[10][NetID (14)][HostID (16)]	80.0.0.0 to BF.FF.FF.FF
Class C	[110][NetID (21)][HostID (8)]	C0.0.0.0 to DF.FF.FF.FF
Class D*	[1110][MulticastGroupID (28)]	E0.0.0.0 to EA.FF.FF.FF

*The address range of E0.0.0.0 to E0.0.0.FF are one hop multicast addresses, and regardless of the TTL, they are only alive for one hop before being discarded.

2.5.2 Broadcast

Unlike unicast which restricts the communication between two nodes, broadcast is a means of distributing information from a single source to all nodes on the local subnet. Broadcast is only available under IPv4 since broadcast was replaced with the more powerful multicast under IPv6. Broadcast addressing is divided into three distinct groups as shown by Stevens [Ste94] and listed in Table 2-2.

Table 2-2 IPv4 Broadcast Addressing

Broadcast Address Type	Broadcast Address Format [Net | Subnet | Host]
Limited Broadcast	[FF.FF.FF.FF]
Net Directed Broadcast	[NetID] [FF.FF.FF]
Subnet Directed Broadcast	[NetID] [SubnetID] [FF]

The Limited Broadcast floods the local subnet when there are multiple hosts interested in the information. This type of broadcast traffic is not forwarded by a router and is restricted to the local subnet. Both the Net Directed and Subnet Directed Broadcasts are forwarded by routers and provide a means to broadcast between subnets located on a LAN. All of these broadcast types are not forwarded in a WAN environment so the traffic does not saturate the Internet. The other common use for broadcast traffic is to request specific services from hosts when all that is known is a port number. The client broadcasts information using the commonly used port for the service and a server responds to the client with a unicast transmission. One obvious problem

with this use of broadcast is every host will receive the packet even if the host does not have a service to handle the incoming packet, adding unnecessary overhead to the host machine.

2.5.3 Anycast

Anycast and multicast share the same characteristics of both unicast and broadcast, but each provides a unique service for routing information. A unique address space for anycast traffic does not exist; anycast addresses are part of the unicast address space. Anycast is used when multiple host interfaces are listening for a particular IP address and the closest interface (as defined by the protocol) will handle the request. The closest interface is usually the one on the nearest path to the requesting node relative to the router [Tay98]. The basic idea is a client is trying request services from identical servers and any server may fulfill the request. There are two different variants of anycast, network-layer anycast and application-layer anycast [Met02]. With network-layer anycast, the network is responsible for selecting the anycast server that services the request, whereas in application-layer anycast, an external process determines which server handles the request. An anycast request will use unicast once a server is assigned.

2.5.4 Multicast

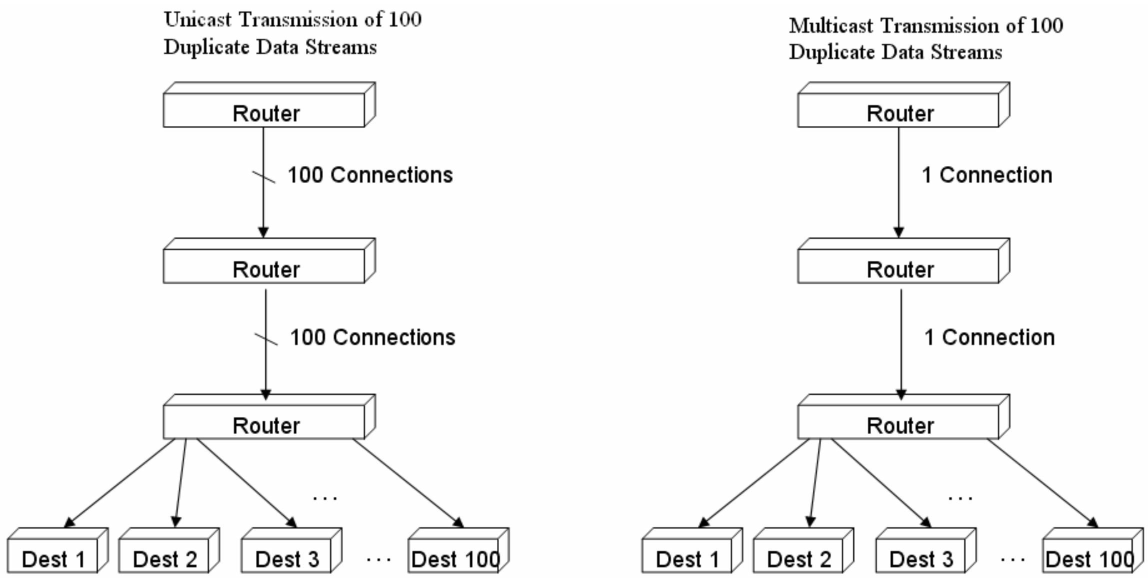
Multicast, as originally envisioned [Dee88], was a flat topology. This soon became unworkable so was modified to a hierarchical model like many other services on the Internet. There are three main ideas in multicast communication [Alm00]

1. IP Style semantics: UDP, a best-effort protocol, is used to transfer information. Group membership does not require an explicit registration or departure request; instead traffic flows from the source when necessary.
2. Open Groups: All that is needed to join a multicast group is the address. Group membership does not have to be known and you don't have to join the group to participate in the multicast, so sources can be outside the group. Additionally, traffic can be received from many multicast groups since there is no restriction on the number of group memberships.
3. Dynamic groups: Membership is volatile – there is no need to explicitly register or send a departure request.

Multicast allows a large amount of information to be efficiently distributed between multiple nodes located in the same subnet. Satellites, because of their large coverage areas, provide an ideal means of implementing multicast [KoF01]. Multicast traffic can be propagated in any of three fashions: one-to-many, many-to-one, and many-to-many [KoF01]. For example, in a one-to-many scenario, a multicast address is assigned to all receivers of a particular multicast group, and a sender simply uses the multicast address to have the information reach all members of the group [DEF96]. Ideally, based on the multicast group membership, there is a single request that flows to the router. This router breaks the stream up into the requisite number of pieces and forwards the packets to all nodes requesting the information. As shown in Figure 2.2, rather than have one hundred unique unicast streams from the source to the destinations, there is a single stream to the local router and then one hundred local streams from the router to the destinations. This saves bandwidth and processing along the path from the

source to the local subnet, and leaves the local subnet area with multiple data streams. These same bandwidth saving concepts can be applied to both the many-to-one and the many-to-many configuration, except that there are multiple nodes sending and either one or many nodes receiving.

Figure 2-2 Unicast and Multicast Comparison



2.5.5 Protocol Variants

There are several proposed schemes for implementing these protocols by combining features from other protocols. For example, multicasting is not inherently supported by the network as multicast aware routers must be present. It is possible to leverage the existing Internet infrastructure to provide multicast like service [PKK01]. This simulated multicasting is done using a combination of unicast and multicast routing algorithms. The client in the local subnet subscribes to a multicast service point. The multicast service point makes a unicast connection to the client. This client becomes the local multicast distribution point called the *feeder* [PKK01]. Other nodes in the feeder's subnet may also be interested in this multicast stream. If so, they send a message to the

feeder requesting to join the group. When the feeder receives unicast traffic from the multicast server, the feeder not only processes the traffic but also broadcasts the traffic onto its local subnet. All clients interested in the multicast can receive the broadcast, effectively creating a multicast environment without requiring multicast routers and other infrastructure changes. This method keeps all multicast transmissions confined to the local subnet and relieves routers along the transmission path and the server from needing to be aware of this multicast group. This simplifies routing tables and removes the overhead of maintaining a multicast tree. Since a broadcast is not routed outside of a given subnet, this protocol is especially appealing since a broadcast is easily done on a LAN.

2.6 Multicast

The ability to use multicast is based on an elaborate set of protocols and multicast aware hardware. While it is possible to implement multicast-like functionality through unicast and broadcast, true multicasting requires multicast-aware routers to manage the flow of information between the sender and the receivers. A multicast aware network requires two key components identified in RFC3170 [QCA01]. First, a client must be able to send the router a join request. This notification should cause the source to send data to the client. Next, the protocol must allow routers to pass routing information between themselves to ensure that all traffic is sent to the subscribed multicast group members.

There are many protocols already in place, as well as proposed protocols, purported to have better multicasting efficiency and support. This study focuses on those

protocols that have been implemented. These can be broadly grouped into two categories: source-based trees and core-based trees [WCA01].

Source-based trees are built from a top (or originating node) down to the receiving nodes. A source-based tree is generally built through a process called flooding. During flooding, the network is saturated with a data packet and it is the responsibility of the router to send a prune request to the server indicating there are no interested nodes in the subnet. Information flows from the source to all receivers. This is a very efficient method if the receivers are densely populated in a given area. If, on the other hand, the layout of receivers is sparse, the overhead from flooding generates excessive and wasted network traffic.

Core-based trees address the problem of a sparsely populated receiver network. A receiver must explicitly ask to join a group by sending the request to a multicast aware router. The router forwards the request to the core router for that particular service, after which membership within the multicast group is established. No flooding takes place, but the network needs to be aware of where core routers are located.

Multicasting can be easily implemented in a LAN environment because it uses a broadcast type network. Therefore, flooding of the local network can be easily accomplished to distribute information. A WAN implements a switched environment where broadcast traffic is suppressed or discarded by routers making multicast more difficult to implement. Broadcast information over a WAN would quickly saturate every router and illustrates that broadcast traffic does not scale well in a WAN environment [DeC90]. One final problem with a WAN implementation is multipath. Multiple shortest paths may exist between any two nodes on a network; and due to variable path

Maximum Transfer Unit (MTU) sizes and latencies, cause unnecessary network overhead [TMH00]. If the path MTU changes each time a routing attempt is made, the benefits of path MTU discovery are negated. Variable latencies may cause packets to arrive out of order. This may result in the initiation of TCP recovery algorithms and the generation of unnecessary retransmit requests.

The goal behind multicast communications is to distribute information from a source node to multiple destination nodes as efficiently as possible. By depicting the nodes as a directed graph, each node in the graph and its corresponding connection is represented by the tuple $\langle a,b \rangle$. If the link connecting the two points on the graph is bi-directional, then $\langle a,b \rangle \leftrightarrow \langle b,a \rangle$. The link may be symmetric or asymmetric, which determines the cost of each direction of communication. Varying link costs complicates shortest path calculation marginally, but the overall idea remains the same. If the links are symmetric, then the directed graph can be simplified to a uni-directed graph. Communication between nodes can be further divided into source specific and group shared [SaM00]. A single node sends and all others receive under source specific communication. With group shared, each node can send and each node can receive. Multicasting, then, can be distilled into the following problem [SaM00]: find a tree T in a graph G, such that T minimally spans all vertices of the multicast group M, yielding the multicast tree. This optimal route spanning tree is called a Steiner tree, and is NP-complete. Solutions are trivial when only two nodes in G are communicating (e.g., unicast) and when all nodes in G are receiving (e.g., broadcast), but otherwise a cost-delay tradeoff must be made to solve this problem in polynomial time.

The earliest multicast implementation was called MBone, which was an abbreviation for multicast backbone [Alm00]. MBone used multicast aware areas of the Internet and tunnels connecting these areas. The program, *mrouted,* was in charge of moving information encapsulated in traditional IP packets between these multicast areas via the tunnels [Alm00]. The topology for this network was flat and not workable for large-scale deployment, leading to the present hierarchical approaches of today.

2.6.1 Internet Gateway Management Protocol (IGMP)

IGMP provides management functions to support multicast traffic and is implemented in the IP protocol stack of the host [Fex97]. The host IP layer is extended to support the multicast extensions and facilitate communication between the host and a multicast aware router. Multicast traffic flows via the Class D address space under IPv4, and under the multicast address range (0xFF in the highest octet) of IPv6. Group membership is managed via IGMP through Join/Leave messages and the status of the group is available through Query/Report messages. The address 224.0.0.1 is reserved to denote all multicast hosts, and the address 224.0.0.2 denotes all multicast routers [Fex97]. A Query sent to all multicast hosts returns information about the current membership and group population. A host returns a report identifying the groups associated with the host. When a host joins a group, it immediately sends out a Join-Group report to ensure the host becomes a member of the multicast group even if it is the first host to join [FiD01]. When a host leaves a group, it immediately sends out a Leave-Group report to remove itself from the multicast list. A Leave-Group report is not sent if a host knows that this is not the last host to reply to the Query, and hence there are other members of the multicast

group still present [Fex97]. The Join-Group and Leave-Group report ensure all multicast routers are aware of the status of the node sending the report.

2.6.2 Distance Vector Multicast Routing Protocol (DVMRP)

The DVMRP is a dense protocol since it assumes that there are many group members in a local subnet. DVMRP builds its trees using a broadcast and prune paradigm. In this paradigm, a packet is broadcast to all routers and if a router has no subscribers, a prune message is returned to the source [Alm00]. Each packet traverses the local network and the router accepts the packet on its incoming interface. The router verifies that the incoming interface is the same interface that would be used for the outgoing traffic to the originator. If the interfaces match, this router is added to the Reverse Shortest Path (RSP) tree, otherwise the packet is dropped and the router is not on the RSP [DeC90]. Once a packet arrives at a leaf router, it is distributed according to the membership specifications that were identified using IGMP [Alm00]. If there are no members for the multicast stream, a prune message is sent for this particular interface identifying the source and the group. If a member attempts to join a group after the prune message has been forwarded, a cancellation of the prune message must be sent to add this router back into the tree [SaM00]. Because the DVMRP is a dense protocol, packets are forwarded over an interface until a prune message is received. To limit the lifetime of a multicast packet, a TTL value may be assigned. The value is decremented by every router handling the packet. The packet is dropped once the TTL reaches zero. A significant disadvantage of DVMRP is state and routing information must be maintained for every reachable node in the network. This leads to potentially large and cumbersome lists of nodes that are interested in multicast traffic [DeC90].

2.6.3 Multicast Open Shortest Path First (MOSPF)

MOSPF is an extension of OSPF that routes multicast information over the presently available shortest path. The multicast router maintains a list of paths and group membership information. The list of paths is based on the usual unicast routing topological information of the router. Group membership information is available via an extended message added to the OSPF protocol. Group membership information is distributed via flooding, while data is only distributed to those nodes that have asked to join the group [Alm00]. When the router receives a unicast packet directed at a multicast group, the router verifies that the node is subscribed before it computes the shortest path and forwards the packet. A shortcoming of this particular protocol is the need to compute the shortest path on demand; this adds routing overhead to every packet.

2.6.4 Core Based Tree (CBT)

A CBT has a core from which all nodes emanate via the shortest path. From the core, various routers are used to send multicast information to specific hosts. Route information is stored only for a specific branch of the tree, greatly simplifying the amount of routing information in the routers. Each multicast group has a single shared tree rooted at a core router. To subscribe to a tree, a Join request is sent to the core router that, in turn, adds the requestor to the correct branch. A shortcoming of this algorithm is traffic is concentrated on a single link [SaM00].

2.6.5 Protocol Independent Multicast (PIM)

PIM is a method of distributing traffic via multicast regardless of the type of underlying unicast routing algorithm. There are two main variations of PIM: sparse

mode and dense mode. The difference between the two is the level of detail that is maintained by routers. Sparse mode requires routers on the path of the multicast group and host to maintain the state of the multicast routing tables. Dense mode requires all routers maintain the state of each host in the multicast group. Dense mode also assumes that there are many receivers densely distributed around the source. Therefore, the network is flooded with packets unless an explicit prune message is received. Sparse mode group members send an explicit join. Consequently, PIM-SM is the better choice when receivers are sparsely distributed around the source.

2.6.5.1 Protocol Independent Multicast-Dense Mode (PIM-DM)

PIM-DM is similar to DVMRP with two major differences [Alm00]. DVMRP generates its own routing tables whereas both PIM-DM and PIM-SM use the standard unicast routing table. Secondly, PIM-DM automatically forwards packets to all outgoing interfaces, unlike DVMRP which does not forward to outgoing interfaces which have failed a Reverse Shortest Path (RSP) check [FiD01]. The continued flooding of outgoing links that have failed a RSP check results in redundant prune messages and wastes bandwidth.

2.6.5.2 Protocol Independent Multicast-Sparse Mode (PIM-SM)

PIM-SM addresses the problems associated with sparsely populated networks and limits unnecessary prune requests [EFH98, FHH02]. PIM-SM is based on the CBT concept so there is a single rendezvous point (RP) per group. During the initialization process and periodically thereafter, the RP's group mappings are transferred between routers to keep the group tables current and distributes topological changes to the

network [FiD01]. To receive multicast traffic, an explicit join is sent to the RP for the multicast group causing the requesting host to be added into the CBT. If there are no members of the group subscribed and the source sends traffic to the RP, the RP will send a stop request to the source rather than continue to let the source forward unnecessary traffic. The RP can degrade network performance when the RP is heavily utilized resulting in a bottleneck, or when a route from the source, to the rendezvous point and receiver is not optimal [Alm00].

2.6.6 Border Gateway Management Protocol (BGMP)

BGMP provides a means of transferring routing information across domains [SaM00]. If a router has group members that belong to a multicast tree, then the router registers itself into the multicast traffic stream. In the same manner, BGMP combines the membership requirements of all subsidiary routers and registers itself into the multicast traffic stream based on the status of the subsidiary routers. BGMP condenses membership information of its many child nodes and presents this membership as a single tree [SaM00]. Management of a BGMP tree is accomplished using Join/Prune messages between various border routers to track which domains are subscribed to a given multicast stream.

2.6.7 Proposed Multicasting Algorithms

There are many proposed algorithms to address actual (and perceived) failings of multicasting over satellites. One proposal, the Datagram Routing Algorithm (DRA) simplifies the dynamic satellite environment and avoids rebuilding the multicast trees as the network topology changes [EAB02].

This algorithm is designed to efficiently create and maintain multicast trees from the source of a multicast group. Protocols, such as those mentioned above, can create non-optimal paths due to satellite movement and the dynamic nature of satellite networks. To simplify the representation of the dynamic satellite environment, each satellite is treated as if it stays in a single logical location. Satellites are constantly in motion around the earth, but based on the constellation layout, each satellite is placed inside an evenly spaced grid. As one satellite passes out of its logical location another satellite is taking its place [EAB02]. Multicast trees are then built using the logical grid and not the physical satellite. The trees are only modified when a member joins or leaves the group. Routing tables are passed to the satellite's incoming successor as the current grid satellite passes out of its logical location. Finally, from a networking perspective, each satellite is always in contact with four satellites: the one in front, behind, left, and to the right, and all paths are maintained in a primary and a secondary direction. These inter-plane inter-satellite link (ISL) connections have fixed propagation times that are maintained by each satellite [EAB01].

Multicast group membership is handled in a strictly logical sense. The grid is a logical location in the overall tree. The satellite presently in that grid is the physical implementer of that node in the multicast tree. Packet routing follows the minimum spanning tree between a source and destination in this logical environment. Since the algorithm is packet-based as opposed to connection-based, a shortest path routing determination is made for each packet and there is no need to maintain a fixed path for a communication link [EAB01]. Path determination is made in three phases: direction estimation, direction enhancement, and congestion avoidance [EAB02]. Each node in the

tree is addressed not only by its logical location, but also according to the four directions. The DRA fills in the direction for the primary and secondary path based on the criteria mentioned for the three phases of path determination. Membership to the multicast tree is handled via three operations: Join, Leave, and Update [EAB02].

2.7 Summary

The literature review in this chapter presents progressively more detailed descriptions of satellite communication and multicasting over satellite networks. After briefly covering satellite dynamics, a section on the low level communications protocols is presented. Next, an overview of four common methods of performing network communication is discussed. This chapter concludes with an in-depth look at various multicasting protocols.

3. Methodology

3.1 Introduction

This chapter describes the methodology used in this thesis. The hypothesis is presented, followed by a problem statement, leading to a description of the methods used for building and running a simulation. Finally, this section of the thesis presents an overview of the statistical methods that are used and how those methods are applied.

3.2 Background

The literature review provided an overview of satellite communications, networking protocol requirements, and various multicasting algorithms. Many of these multicasting algorithms have been applied successfully to terrestrial communications and are postulated to work in a similar fashion for satellite communications. Contrary to the relatively stable network configurations present on Earth, satellites with their rapid mobility and limited processing capabilities present unique challenges requiring further investigation.

The rapid movement of satellite systems affects the speed at which decisions and tree updates must flow, but due to the predictable nature of a satellites orbit, it is possible to forecast the satellite topology at any given time. The ability to accurately locate a satellite, either in the present or in the future, can be exploited to remove much of the uncertainty of where the satellite node is for a multicasting communication network.

Satellite-based multicasting presents a unique opportunity to exploit the capabilities provided by multicasting algorithms. Extending terrestrial multicasting research and applying it to space based assets is a logical step from the current unicast method currently used in satellite-based networks.

3.3 Problem Definition

Satellite multicasting is the focus of various current and proposed research efforts [Pra99, Tho01, Fos01]. This research extends prior work and incorporates Protocol Independent Multicasting-Dense Mode (PIM-DM). Prior research focused on the system performance of a six plane, 66-satellite LEO constellation using ODMRP or DVMRP. This is extended to include the PIM-DM protocol.

3.3.1 Goals and Hypothesis

Of the three protocols under investigation, DVMRP and PIM-DM assume there are many recipients for their multicast traffic and flood the network with packets. The ad-hoc multicasting protocol (ODMRP) operates over a topologically dynamic network where nodes are constantly in motion and only base stations are stable

Although there are a fixed number of satellites in the network, there may be hundreds of users of a multicasting service. The large number of users coupled with a fixed number of satellites generates a range of possible user configurations, from sparsely to densely populated. The goal of this research effort is to determine the system performance of PIM-DM and compare this to the already defined models of DVMRP and ODMRP.

ODMRP accommodates the inherent mobility of each node through a mesh-based structure. The mesh is built as required by the protocol and responds to intermittent activity and rapid topological changes better than a tree. Unlike a tree-based approach, the mesh exploits possible redundant paths between nodes to route traffic. The advantage of using this type of protocol is its ability to adapt to additions to the network and have multiple routes between nodes. Disadvantages stem from the need to inform other

participants of the mesh of any additions or deletions to the network, possibly leading to excessive messaging requirements for topology updates.

DVMRP is a dense protocol and floods the network with packets, only removing a branch when a Prune message is received. Tree branches extend from the source to all destination nodes and routes are determined via a Reverse Shortest Path (RSP) algorithm. Due to continual flooding (until a Prune request is received), startup network utilization costs for a DVMRP tree are high. DVMRP should provide efficient transmission capabilities after receipt of all prune requests since traffic is limited to nodes interested in the multicast stream. State and routing information for every node involved in the DVMRP multicast session must be maintained. It is expected that higher subscriber loading levels will lead to decreased performance and increased processing time.

PIM-DM is a dense mode protocol similar to DVMRP except the protocol does not maintain routing tables; instead, the underlying unicast transport mechanism maintains them. Unlike DVMRP, a RSP check is not performed causing a packet to be forwarded out all interfaces until a Prune message is received on the corresponding interface. While less information is maintained in PIM-DM and it requires less processing requirements, the underlying network utilization is expected to be higher than DVMRP due to the absence of the RSP determination on the outgoing interface.

It is expected that PIM-DM will perform best in a LEO satellite multicasting environment. The draft specification was followed for implementation [ANS02] so the model takes advantage of aspects of the protocol that were redesigned to address the original shortcomings of PIM-DM. One of the additions is the capability to maintain the state of the current network through the use of a State Refresh message. By using a State

Refresh message, it is not necessary to re-prune links that timeout and reactivate. This will increase the data-to-overhead ratio since overhead is minimized for active connections.

3.3.2 Approach

The approach to testing the hypothesis is to inject a known workload into the system. The effects of this workload will be measured through three metrics: data-to-overhead, receive-to-sent, and end-to-end delay time. The metrics will be compared to prior research for DVMRP and ODMRP giving a performance analysis comparison between the three protocols.

3.4 System Boundaries

The System-Under-Test (SUT) is the 66-satellite LEO constellation and the corresponding ground stations. The Component-Under-Test (CUT) is the multicasting algorithm being investigated. Users access the system through the ground stations which send packets to the satellite system en-route to the receiving system. A LEO satellite located at an altitude of approximately 7,000 km gives the ground station an in-view time of approximately 15 minutes. Therefore, all communication events last no longer than 10 minutes and are complete before a satellite moves out of the ground stations field of view.

By defining the SUT boundary to include the ground stations, all users accessing the system logically prior to the ground stations (e.g., Internet) are not part of the system. All multicast messages originate at the ground stations, traverse a route on the satellite network, and end at another ground station. Implementation details such as

subscriptions, routing, and tracking of each satellite are modeled inside this network based on generated traffic flow.

3.5 System Services

The goal of the multicast service is to deliver packets from one or more senders to one or more receivers. Delivery occurs using a multicasting algorithm regardless of the type of data. Although multicasting algorithms are implemented differently, all provide the same basic set of services and ultimately the intent is to successfully transmit a packet to the destination. The outcomes specified in Table 3-1 are derived from the outcomes in [Tho01].

Table 3-1 System Services Outcomes

Outcome	Recipient	Contents
Successful Delivery	Correct	Correct
Incorrect Delivery	Incorrect	Ignored
Incorrect Contents	Correct	Incorrect
Failed Delivery	Unknown	Ignored

3.6 Performance Metrics

Based on the system services, the result of transferring a packet can be either successful or unsuccessful. Following a successful or unsuccessful outcome, the metrics gathered are data-to-overhead, receive-to-sent, and end-to-end delay time.

The data-to-overhead metric indicates how many overhead bits are required to transmit data bits. A certain amount of overhead is required to perform normal network operation, but excessive overhead wastes available network bandwidth. Ideally, this ratio should be close to one indicating that the majority of the data flowing over the network is data.

The number of packets received is not useful by itself, but combined with the total number of packets sent, the efficiency of the multicast algorithm can be determined. The received-to-sent ratio is number of packets received compared to number of packets transmitted (p_{recv}/p_{tot_tx}). This ratio is a simple means of portraying how efficient a system is. Even though a lost or dropped packet could be due to an external event, ultimately this is a failed delivery. As this ratio approaches one, the model approaches completely efficient transmission and reception. The number of lost (e.g., failed delivery or dropped) packets is determined by calculating the difference between the number of packets successfully received and the total number of packets transmitted.

End-to-end delay time is how much time is required for the packet to traverse the system. The time required to receive the packet via the network card and transmit the packet out of the network card is excluded from this processing time as this transmit/receive time is outside of the system's control. While each multicasting algorithm may use a different route, the most efficient route is the shortest route with the least number of hops and minimal network congestion along the nodes.

3.7 Parameters

The parameters for the SUT are divided into two categories: system and workload. The system parameters are those that define the underlying system model and stay constant between simulation runs. The workload parameters are those characteristics that affect the behavior of the workload. In order to determine the sensitivity of each parameter, Principal Component Analysis (PCA) is used to verify parameters are correctly identified and that those parameters contribute sufficiently to the overall

variance [Jai91]. A parameter can be eliminated (and possibly replaced if a better parameter is found) if it contributes minimally to the variance of the system.

3.7.1 System

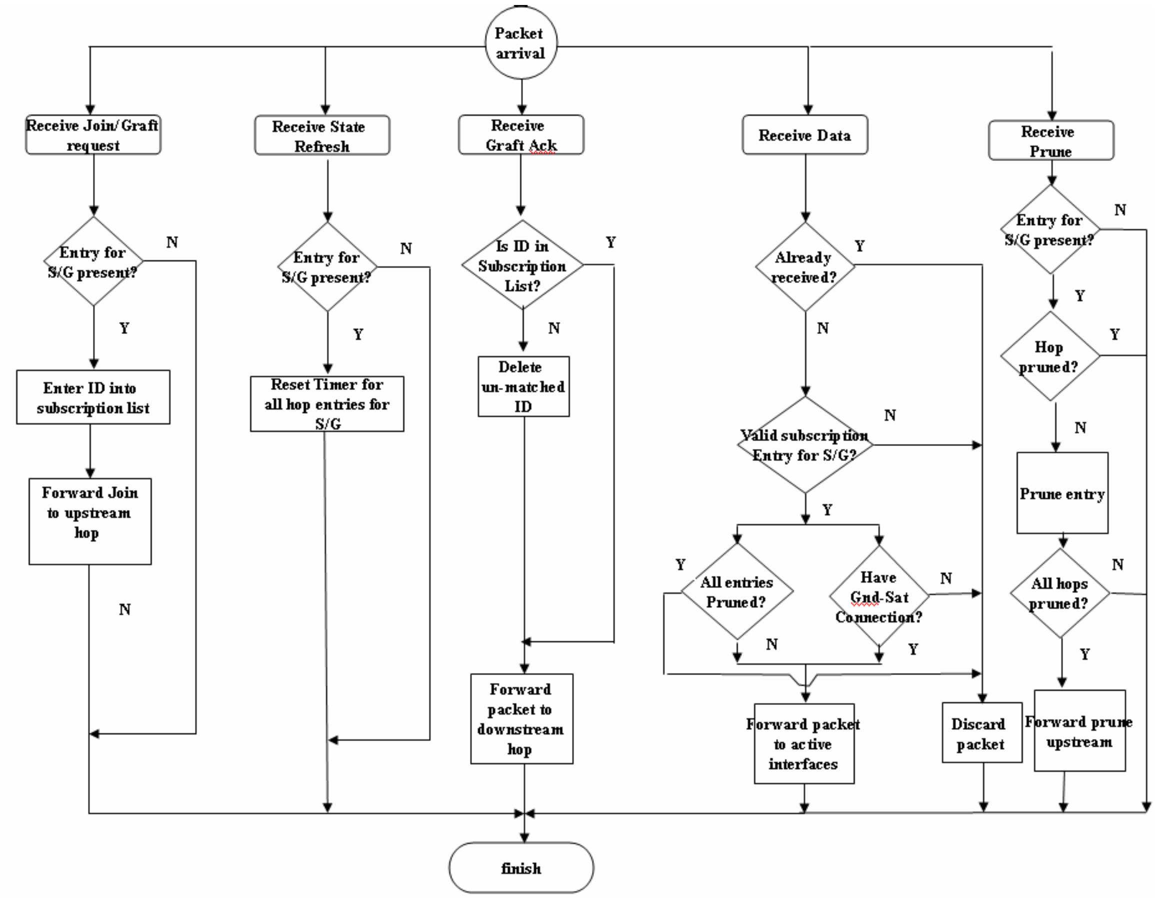
The satellite system is based on the Iridium satellite constellation. The network consists of 66 satellites divided into six planes with 11 satellites per plane. The satellites are placed into co-rotating planes 31.6 degrees apart and counter-rotating planes 22 degrees apart [Tho01].

Each satellite in the 66-satellite constellation is further defined by the number of users and network bandwidth at each satellite. A satellite has 48 spot beams, can handle 80 users per cell, and provides 2.5 Gbps of available bandwidth on the up/down/inter-satellite links [Tho01].

The third system parameter is queue service time. Queue length is dependent on rate of arrivals and service times. Packet arrival times are exponentially distributed, leading to an M/G/1 queuing system. Infinite queue lengths are assumed at the processing nodes to identify bottlenecks in the multicasting system.

The final system parameter is the type of multicasting algorithm used (PIM-DM). Figure 3-1 presents a overview of how PIM-DM is implemented in the simulation. The satellite provides routing services for datagram packets based on the configuration setup by the PIM control packets.

Figure 3-1 PIM-DM Protocol Flowchart



3.7.2 Workload

The number of packets per second introduced into the system defines the workload of the system. Packet generation times are exponentially distributed to facilitate comparison to earlier work [Tho01]. Packet inter-arrival times are categorized as one of three loading levels: low utilization, medium utilization, and high utilization.

The packet length includes both header and data bytes and is included in the total packet size. Header information is constant (40 bytes) and includes an IP header (20 bytes), and a TCP header (20 bytes). Based on data gathered by the NASA Ames Internet Exchange (AIX), average packet sizes are approximately 400 bytes with a standard deviation of 500 bytes are also common [McC00]. Packet sizes of 40 bytes

(minimum TCP packet length), 576 bytes (minimum amount that must be transmittable without using Maximum Transfer Unit [MTU] discovery) and 1500 bytes (maximum size of an Ethernet packet) [McC00] are used.

The number of multicast groups available is determined by the loading level. Group membership spans 5, 10, 15, 20, 30, and 40 and are assigned to source and subscriber nodes to generate the various loading configurations. Multicast protocol efficiencies (or deficiencies) are best illustrated by this assignment approach, and will isolate any problems that are present with specific initial conditions. Possible items that affect the multicast group behavior are: starting subscription node(s) and locations, number of groups, number of subscribers, and satellite constellation location.

Finally, based on a single multicast group, group membership is expected to stay constant. The number of senders and receivers assigned to a multicast group does not fluctuate, since node failure is not being considered. To investigate multicast protocol behavior, two communication configurations are modeled: one-to-many and many-to-many. This sender/receiver ratio is based on one of three loading levels: low, medium, and high.

Parameters of the system and workload are summarized in Table 3-2.

Table 3-2 System and Workload Parameters

System	Constellation size	66 satellites
	Number of users/satellite	3840
	Up/Down/Inter-satellite link speeds	2.5 Gbps
	Queue length	Infinite
	Queue Service Time	Exponential
	Multicasting Algorithm	PIM-DM
Workload	Packet Length	$\mu = 400$ bytes, $\sigma = 500$ bytes
	Multicast groups	1 to 40, depending on configuration
	Density	Sparse/Dense
	Transmission Method	One-to-Many, All-to-All

3.8 Factors

There are two system factors and one workload factor under consideration. The two system factors are queue service time and multicasting algorithm. Queue service time is how quickly the packet is removed from the queue and sent. Based on Little's Law, the utilization, $\rho = \lambda / \mu$, must be less than one otherwise the system is unstable. By modifying this factor, potential instabilities with either the network, loading level or algorithm can be identified. The multicasting algorithm is limited to the system under investigation, mainly PIM-DM.

The two workload factors are density and transmission method. The density affects the number of senders and receivers and how the multicasting algorithm performs. Sparse density has the total number of groups spread in a round-robin fashion across all receiving nodes. Dense has each receiving node join all possible groups that are generated by the source. The final factor is transmission method which is modeled as one-to-many or many-to-many.

Factors of the system and workload are summarized in Table 3-3.

Table 3-3 System and Workload Factors

System	Queue Service Time	Exponential
	Multicasting Algorithm	PIM-DM
Workload	Density	Sparse/Dense
	Transmission Method	One-to-Many, All-to-All

3.9 Evaluation Technique

The satellite networks under investigation have not implemented a multicasting algorithm to validate the results of the simulation against. The current research effort is

being used to assess a "what-if" scenario, so the type of evaluation is simulation. The correctness of the modeled satellite network is validated based upon the underlying unicast network model, since unicast is the current means of performing satellite communication.

PIM-DM is integrated into the satellite network model and its performance analyzed. PIM-DM results are also compared to the DVMRP and ODMRP performance measurements.

3.10 Experimental Design

The experiment uses the satellite model defined by [Tho01], which specifies the layout and position of each satellite in the 66-satellite constellation. This model also defined the workload based on three loading levels (packets/second) and specified the number of receivers based on two sequences of three levels.

In writing the code necessary to implement PIM-DM in OPNET®, the RFC's or draft specifications which define PIM [DEF99, ANS02] are used to ensure accuracy of the model. The results are compared to terrestrial results to validate the models accuracy and behavior. Any simplifications introduced to make the modeling more efficient or remove unnecessary functionality is documented. As long as the functionality being removed does not affect the performance of the protocol, the model is simplified to facilitate more rapid simulation.

After correctly implementing a functional PIM-DM multicasting algorithm into the modeled satellite system, the experimental phase begins. Comparisons are based on a proportional confidence interval (90% CI) to determine which algorithm performs the

best. Based on the stated factors, a full factorial experiment would require the number of experiments shown in Table 3-4.

Table 3-4 Experimental Design Determination

Transmission mechanism	**Loading level**	**Ground Node location start point**	**Multicast group members**	**Density**	**Runs for CI**		**Total experiments***
1-*	3	7	6 (5,10,15, 20,30,40)	2 (Sparse, Dense)	3		756
-	3	7	3 (5,10,15)	1 (All)	3		189

*Queue Service Time, Multicast Algorithm, and Packets/second had a value of 1 so did not contribute to the total number of experiments (and therefore were not represented in the above table)

3.10.1 Scaling

Scaling of data packets to emulate increases in the load on the system was used to generate higher loading levels for a given level of packet transmissions [Tho01]. Without scaling, generating higher loading levels under OPNET® meant sending more packets – but higher packet levels requires greater overhead, processing power, and physical memory. Scaling provided a convenient method to increase the size of the packet by incorporating multiple packets into a single packet. OPNET® only had to transfer one packet on the simulated network and the statistics could be adjusted to correctly count that one packet as the actual number of packets it represented.

The End-To-End (ETE) delay is adjusted to account for scaled operations by adding an additional "delay" component into all data packets which accumulates the delay as the packet traverses the network. Once the packet arrives at the receiving node, the "delay" component is extracted from the packet and incorporated into the ETE calculations.

The ETE for a data packet is the cumulative time the packet spends in the network as the packet travels from the source to the subscriber. The average time is

$$ETE = \sum_{i=1}^{n} T_{av_i} + \sum_{i=1}^{n-1} d_i \qquad 3.1$$

where n is the number of hops along the route, T_{av} is the length of time the packet spends at each hop, and d is the computed propagation time between sequential nodes along the path of the packet.

The time in the system of a non-scaled data packet is

$$T_{av} = E[s] + \frac{\rho F E[s](1 + C_s^2)}{2(1-\rho)} \qquad 3.2$$

where T_{av} is the length of time the packet spends at each hop, $E[s]$ is the expected service time, ρ is utilization, F is the scaling factory, and C_s is coefficient of variation of the service time.

The scaled version of the time in the system is

$$T_{av}^{'} = FE[s] + \frac{\rho F^2 E[s](1 + C_s^2)}{2(1-\rho F)} \qquad 3.3$$

and the scaled ETE is

$$ETE^{'} = \sum_{i=1}^{n} T_{av_i} F + \sum_{i=1}^{n-1} d_i F \qquad 3.4$$

Scaling of packets is incorporated into the PIM simulation to drive loading levels and determine how the system behaves under stress. Interestingly, pilot studies show that scaling does not produce the intended effect using PIM, namely the ability to increase the load on the system. This is due to how a "packet" is defined. In the studies of ODMRP and DVMRP, the data packet had the overhead (the routing information) included in the packet [Tho01]. PIM is implemented with separate routing and data packets. Therefore, as packets are scaled, both the data and routing packets were scaled increasing the routing

overhead in lockstep with the amount of data transmitted. Because PIM has separate data and routing packets, and increase in the scaling only increases the efficiency of the overall protocol since the routing information stays constant.

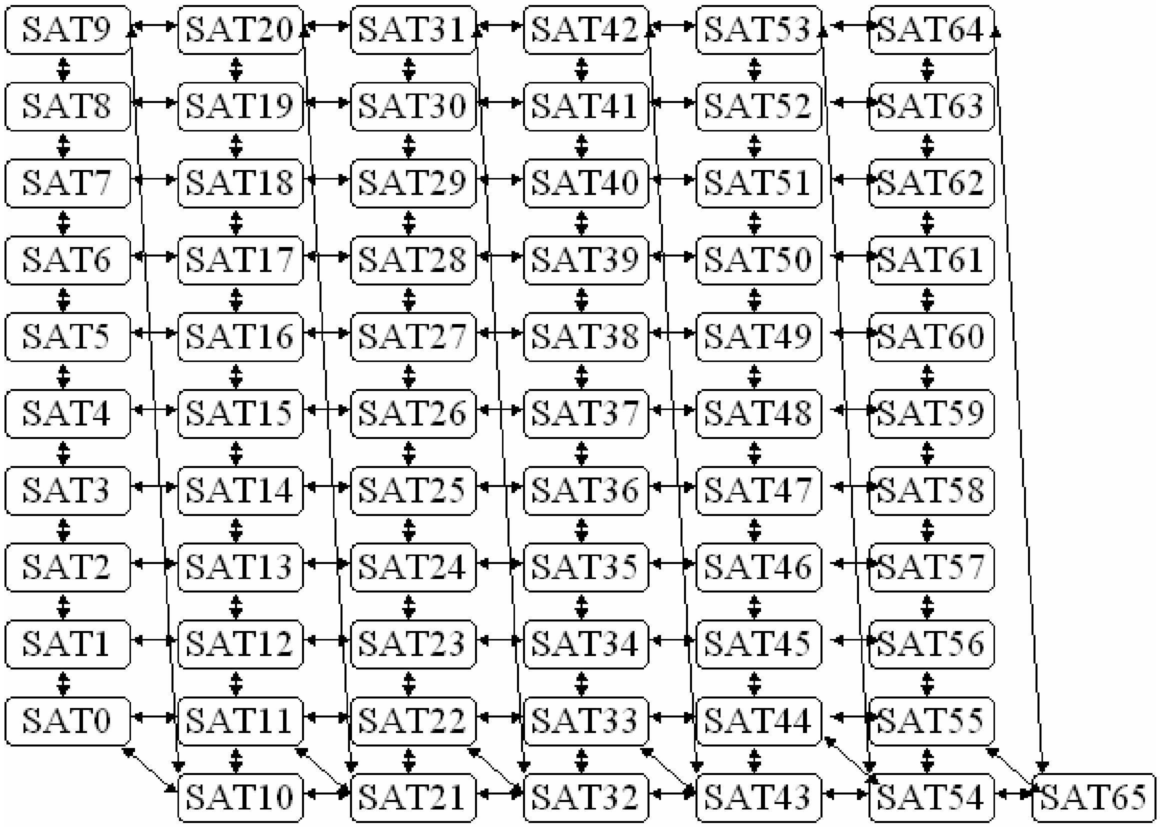
3.11 Implementation Details

Implementing a complex protocol in a discrete event simulator requires that assumptions be made and parts of the protocol not implemented be documented and explained. The PIM-DM protocol was not implemented exactly as described in the specification. PIM-DM terrestrial implementations do not have the rapid mobility of a satellite network and the constant adjustments this mobility forces on the protocol. Network links are instantiated at satellite nodes based on the current shortest path which may change within a matter of seconds. The main changes made in the simulation model are a result of the requirement to tear down and re-build active satellite connections to ensure the packets use the shortest path.

3.11.1 Satellite Network

The satellite network shown in Figure 3-2 is identical to the network used in prior studies [Pra99, Tho01, Fos01]. The satellite network is in a stable configuration, barring catastrophic satellite node failure. Each satellite has a left, right, forward, and backward neighbor. The one exception to this occurs at the counter-rotating seam where satellites 0 to 9 and satellites 55 to 65 are going in opposite directions and communication is not possible.

Figure 3-2 Satellite Network Logical Connectivity



The static neighbor paradigm simplifies route optimization since all distances between adjacent satellites are a single hop, regardless of the actual physical distance between satellite nodes. While neighbor satellites are static, routes constantly change as satellites move through their orbits and become the closest node to either a neighbor satellite or a ground node. In a departure from prior work [Tho01], the prohibition of satellite communication to left and right neighbors across latitudes of ± 60 degrees is abandoned. The rationale for this prohibition is that a satellite outside this latitude window was converging rapidly with other orbits so communication was not possible. This implementation of PIM-DM removed this restriction since the routing protocol only allows a single route from a source to a destination. A single route introduces the possibility that a packet could flow one way and have no way to return.

A satellite can have multiple connections to various locations on earth within its range of communication. However, we assume a single ground node is the closest neighbor for a satellite. Multicasting is extended so that the earth node became a router as well and all communication to entities close to the earth node are aggregated and sent as a single data stream to the earth node before being sent to the various subscribers.

The slant range distance of the satellite is calculated using the Pythagorean Theorem and is the length of the hypotenuse between the ground station and the satellite, or 2,342 kilometers.

3.11.2 Routing Protocol

The routing protocol implemented is a modified form of RIP. The protocol consists of three main message types shown in Table 3-5.

Table 3-5 RIP Message Types

RIP Message	Description
Probe	Sent by each satellite router to all neighbor satellites to determine if any changes to the network have taken place.
Report	As changes are detected in the network, a report is sent to update each neighbor with the sending satellites routing and hop information.
Ground	Sent by the ground node to update the satellite information on the current ground to satellite hop. The packet is sent to the new satellite as well as the old satellite to ground neighbor.

The Probe and Report messages work together to maintain the state of the satellite network tables. Probes are periodically sent to all surrounding neighbors to identify network topology modifications. Once a change has been found or the report time expires, each satellite broadcasts a Report to all adjacent neighbors. The report contains a complete list of the sending satellites routing tables. The recipient of the report message

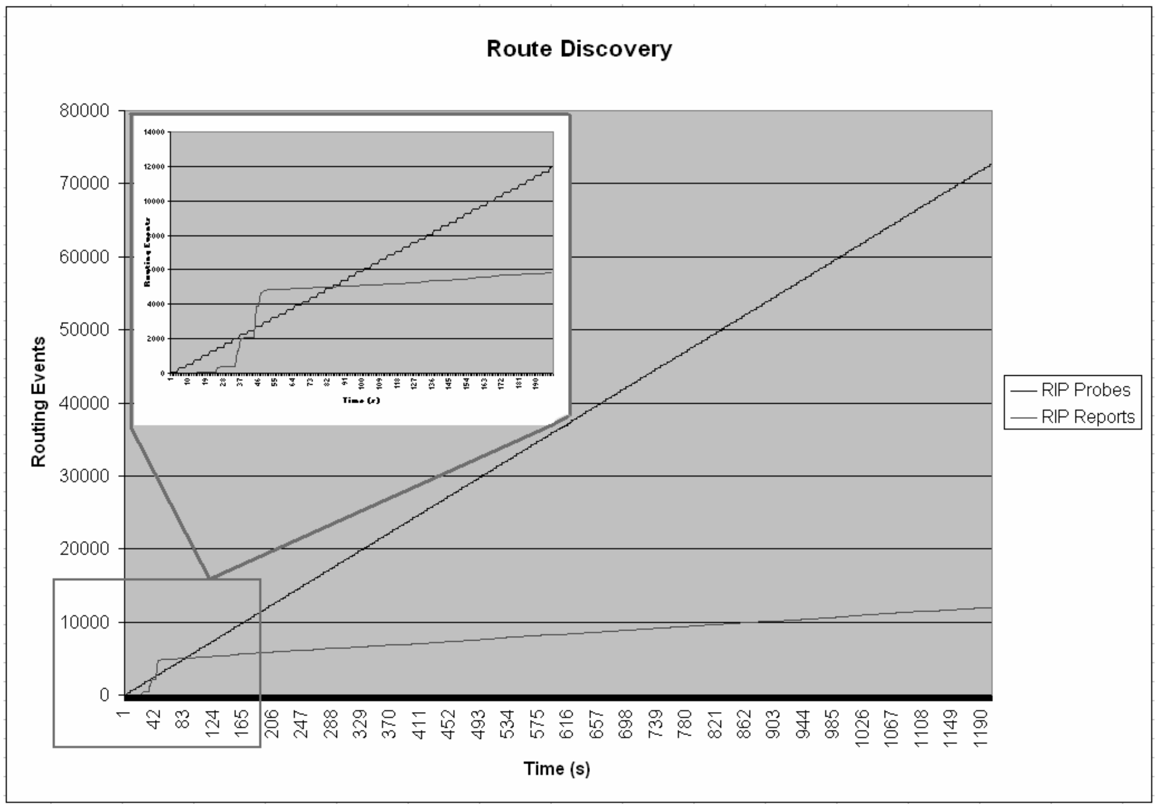
parses the information and updates its routing tables (as required) if a shorter path is available.

The Ground message facilitates satellite handoffs at the communication end points. The ground station is responsible for calculating the closest satellite to itself at one second intervals. The results of this calculation determine whether a handoff is necessary to the new "closest" neighbor satellite. If a handoff is required, two Ground messages are sent, (1) the losing satellite to relinquishes its tasks as the ground to satellite connection and (2) the gaining satellite to accept the task of being the ground to satellite connection. These two messages cause a cascade of routing messages as routes are adjusted to the ground node end points. To prevent packet loss while the new routes are built, packets are forwarded temporarily from the losing satellite to the gaining satellite. Finally, a PIM Assert message is generated to recalculate the route from the subscriber to the source.

The Probe and Report messages are the most prevalent packets sent and account for the majority of the transmissions as shown in Figure 3-3. Route discovery stabilizes after approximately 100 seconds as shown in Figure 3-3. Discovery generates large numbers of Report messages as all routers are trying to build a complete routing table. Once the tables are built, the number of reports required to maintain the routing tables drops considerably and stabilizes. Figure 3-3 clearly illustrates the initial update when all tables are in transition and current routing information propagates through the network. After initial route discovery, routing reports and probes settle into linear patterns as shown in Figure 3-3 and do not have any further spikes as the protocol maintains the

present state of the network. This particular example illustrates 1200 seconds of simulation time, but the slopes of the lines stay the same for a complete simulation run.

Figure 3-3 Route Discovery



3.11.3 Ground Nodes

The location of the ground nodes matches earlier work and ensure that each geographic region of the earth had a single ground node. The ground node locations are presented in Table 3-6.

Table 3-6 Geographic Ground Node Locations

Ground Node	Longitude	Latitude
Rio de Janero	-43.22	-22.90
Melbourne	144.97	-37.80
Kansas City	-94.59	39.13
Dharan	50.00	27.00
Beijing	116.47	39.90
London	0.00	51.29
Capetown	18.37	-33.93

3.11.4 PIM Protocol Messages

PIM-DM, or Protocol Independent Multicasting – Dense Mode, is the basis protocol for this work. The protocol used in simulation is based closely on the draft specification [ANS02] but changes were necessary to support PIM-DM in a non-terrestrial configuration. Most importantly, to facilitate lossless communication, a means of transitioning a satellite's subscribers had to be incorporated into the network. The fluid nature of the network topology combined with static ground nodes meant that the ground nodes had to instantiate the hand-offs between the ground to satellite connection followed by an Assert to rebuild the link. The RIP Ground message, while not a PIM messages is essential to provide this transition capability.

Timing considerations are adapted with little modification from the draft specification. PIM-DM is implemented using timers and interrupts and adapts based on the various packets received [ANS02]. Table 3-7 summarizes the various timing values used in the simulation and a short summary of what each timer accomplishes.

3.11.4.1 PIM_HELLO

The Hello message is implemented according to the draft specification and notifies satellite neighbors of the existence of a PIM-aware satellite router. As the network is fully composed of PIM-aware satellites, all satellites send and receive Hello messages. The Hello is crucial for building and inactivating neighbor links; no response to a Hello query causes a link to become inactive. The data from the Hello messages is cross checked with the hop lists of the Source/Group entries downstream to verify that active routes are indeed present.

Table 3-7 PIM-DM Timers and Values

Timer Name	Timer Value (s)	Description
RIP Probe Expire Period	30	Time before the routing data contained in a Probe message is discarded unless an update is received
RIP Probe Timer	4	Time between subsequent Probe messages
RIP Report Timer	10	Time between subsequent Report messages
PIM Assert Time	180	Time after last Assert before Assert information is expired
PIM Assert Override Timeout	0.75	Quiet period where the override bit of the assert packet is ignored
PIM Assert Timeout	5	Quiet period where asserts for a given S/G are ignored, unless override bit is set
PIM Graft Retry Period	3	Time after sending a Graft before a GraftAck should be received, else retransmit Graft
PIM Hello Period	30	Time interval for subsequent Hello messages
PIM Initial Send Time	60	Initial time added to random time for initial Broadcast cycle
PIM Initial Subscribe Time	80	Initial time added to random time for initial Join request(s)
PIM Refresh Interval Timer	10	Time before State Refresh messages are sent from source router
PIM Source Lifetime	210	Time interval after receiving last multicast packet that State Refresh messages will be sent
PIM Triggered Hello Delay	5	Upper bound for random delay to send Hello response

3.11.4.2 PIM_STATEREFRESH

The State Refresh message closely follows the draft specification. A change to the specification was necessary to correctly perform routing. The draft specification called for the routing to take place according to the received Hello messages. The

specification also ensured all satellites received a State Refresh message and due to the small size of the network, the State Refresh was broadcast to everyone regardless of Hello status. Through the use of a broadcast, duplicate packets were avoided and complete network coverage was assured. The State Refresh was used to maintain the current status of the PIM network. That network was created by resetting timers that would otherwise cause an already pruned link to become active. This message type is an addition to the draft specification and contributes greatly to the ability of maintaining the current tree efficiently.

3.11.4.3 PIM_PACKET

The Packet message is a data packet routed according to the current entries in the Source/Group list. If a node receives a data packet and is not on the Reverse Path Forwarding (RPF) entry back to the source or does not have any subscribers, then the node must send a Prune message. The one exception is an active Source/Group list that fails the RPF check. In this case, the packet is forwarded to the next hop and an Assert is sent to the requester to notify that the route needs to be rebuilt. This caveat allows for the data stream to continue flowing down a "bad" path until the new shortest path is built.

Occasionally a duplicate packet is found based on the sequence number and the upstream source address of the packet. A duplicate packet is suppressed at the receiving satellite and no further action is taken related to that packet, other than to destroy the packet.

3.11.4.4 PIM_JOIN/PIM_GRAFT

The mobility requirements dictated by the fluid nature of the satellite network required changes to the draft specification to correctly implement these two messages. To differentiate between a ground-to-satellite and a satellite-to-satellite connection, both a Join and Graft are necessary. According to the specification, the Join is only used when there are two nodes below a router and one router sends a prune. A condition where the specification's Join would apply is not possible in this satellite configuration as each satellite has at most one neighbor on each interface. Therefore, the Join is used in this implementation to indicate a ground node is subscribing to a S/G entry maintained by the closest satellite neighbor. The Graft message propagates a Join request through the network to the source router.

3.11.4.5 PIM_GRAFTACK

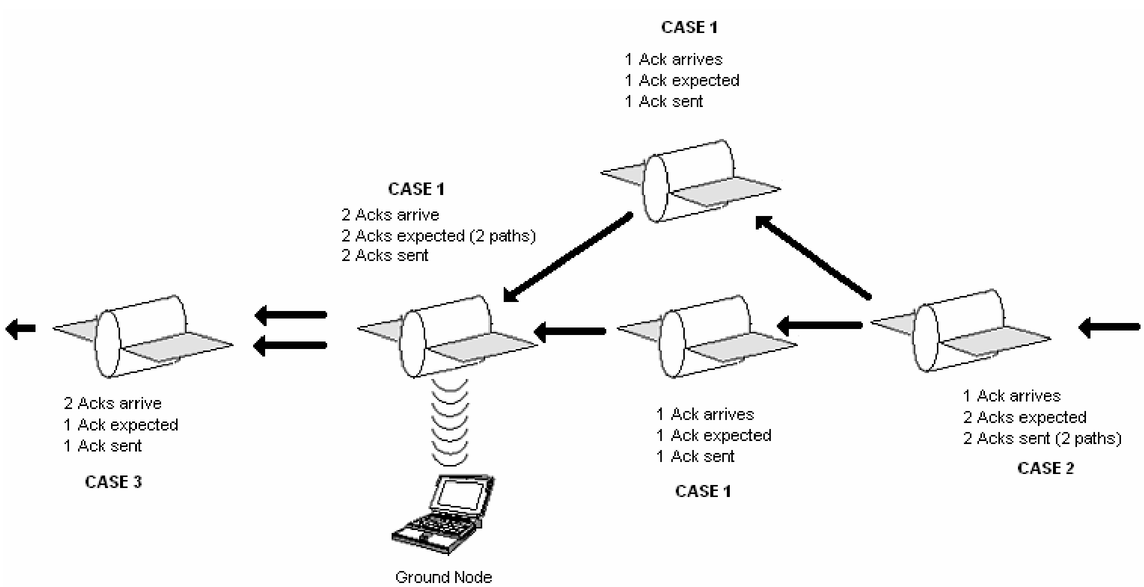
The GraftAck (Ack) message is the acknowledgement sent by the ground-to-satellite neighbor for the source ground node in response to a Graft request. As with the Join/Graft message, the Ack is loosely based on the specification and is modified to be suitable for the satellite environment. The Ack not only acknowledges the Graft request but also tears down the old connection and validates the new connection. The double duty of the Ack message was implemented to reduce message traffic.

Unlike other PIM messages, the number of Ack messages received on the return path is not constant. This makes an Ack message the most difficult PIM message to process. The Join/Graft builds a new link between the subscriber and the source. This

new link may have followed a completely or partially different path than previous requests. Therefore, the path may split, remerge, and split again based on the current shortest path. Nodes along the path processing the Ack have to intelligently determine the correct action and how many Ack messages should flow down each subsequent path.

Incorrectly sending the wrong number of Ack messages can inadvertently prune a node – removing an active link from the multicast tree. Consider the cases shown in Figure 3-4:

Figure 3-4 Graft Acknowledgement Examples



1. All Acks are successfully received, process the Ack.
2. Only one Ack is coming from the upstream hop, but two are needed. Process the received Ack twice. This simulates the receipt of two messages.
3. Two Acks are received, but only one Ack needed. Ignore the duplicate Ack and only process the first Ack.

In all cases, after processing the Ack message, the cleanup phase begins and the old route is removed. The necessary number of GraftAck messages is calculated and forwarded for each active link.

3.11.4.6 PIM_ASSERT

The Assert message is derived from the specification but the specified functionality was not used. The specification uses the Assert message to choose between alternate routes to force a specific route configuration, or to determine which node should be the forwarder for the group [ANS02]. The model uses the Assert to build a new route since the current route has changed due to satellite movement. Therefore, the Assert is accomplishing the same basic function as the specification, but the approach is unique to this satellite network model.

The Assert message is one of the most critical protocol messages since it readjusts routes based on topographical changes. A satellite, failing an RPF check, or a ground node changing upstream neighbors, sends an Assert and the route is rebuilt. Route rebuilding has a two-fold purpose – it ensures the current communications channel adjusts to the new shortest path and it tears down the old route. This reduces but does not eliminate the possibility of duplicate packets. The dual purpose of the Assert is used to avoid the introduction of another message type, although this dual-purpose behavior is not in the specification.

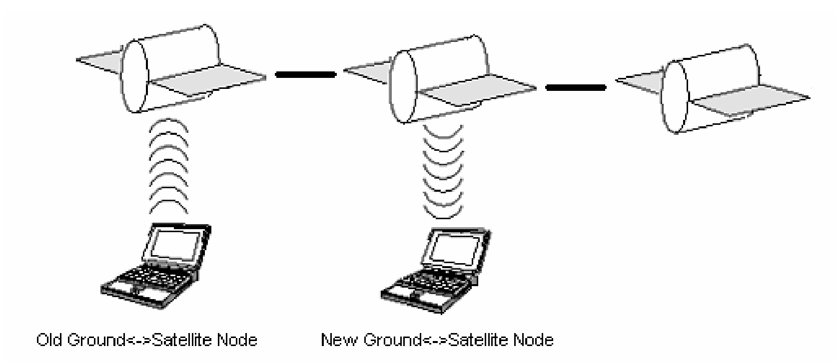
An Assert message that arrives while a prior Assert message is processing is generally ignored. The Assert message that arrives in this timeout window is ignored because multiple Assert messages for the same S/G pair processing concurrently in the system causes congestion and repetitive link regeneration. An Assert must be forced

when a satellite ground node hand-off occurs. Once the hand-off takes place, the new route must be available so an *override* bit is set on the Assert message to force a re-joining for the ground node.

Ground-to-satellite communication is controlled through a simple messaging system. The ground station calculates the distance between itself and all in-range satellites every second. As satellites move overhead, this distance changes and the in-sky neighbor updates accordingly. If a transition is required, the losing satellite receives a RIP_Ground message with the *remove* bit set and the gaining satellite is sent a RIP_Ground message with the *add* bit set. Both satellites process this message and update their routing tables to reflect the new route to the ground. Additional processing is triggered by the *remove* which causes the losing satellite to continue to forward packets to the gaining satellite. A hand-off between two satellites has four situations to consider.

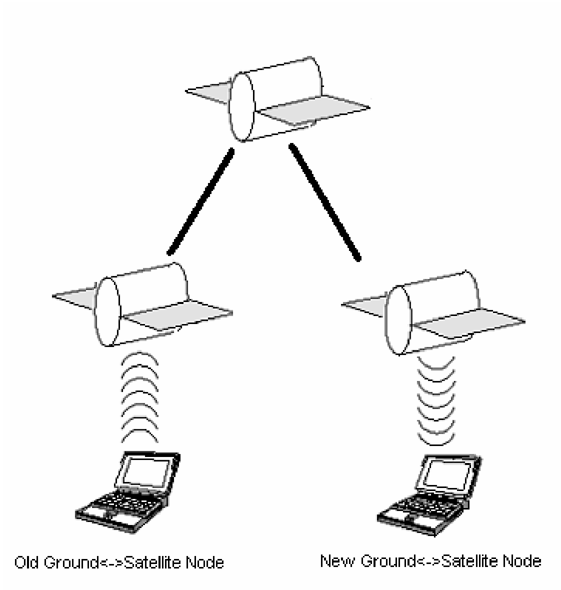
The first occurs when the gaining satellite is still on the path to the source. In this case, the satellite remains part of the already established S/G communication stream. This is shown in Figure 3-5.

Figure 3-5 Handoff to Upstream Neighbor



The second situation occurs during a hand-off to a left or right neighbor as shown in Figure 3-6. The losing satellite removes its S/G entries and forward a Graft request with the *sat_transfer* bit set to the middle satellite.

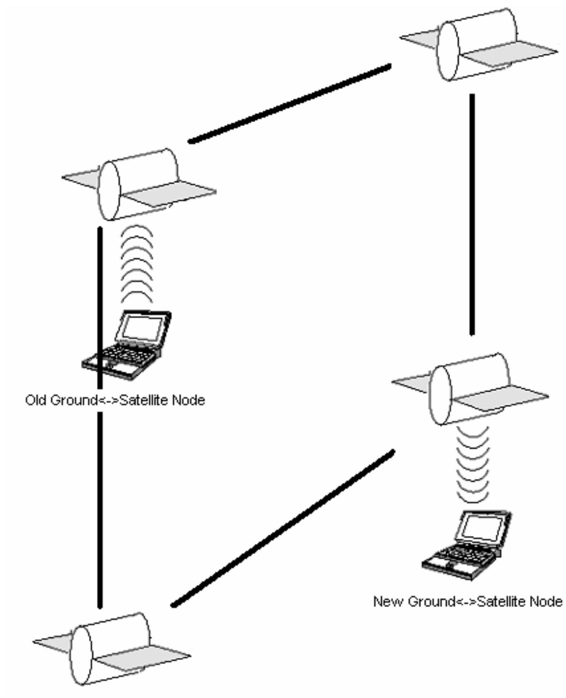
Figure 3-6 Handoff to Left/Right Neighbor through Intermediate Hop



The middle satellite changes its tables to point to the gaining satellite and then forwards the Graft request to the gaining satellite. Finally, the gaining satellite grafts itself in.

The third situation requires an intermediate hop to facilitate successful handoff between satellites. This condition is caused by a transition diagonally across a satellite “square” and is illustrated in Figure 3-7.

Figure 3-7 Handoff to Diagonal Neighbor through Intermediate Hop

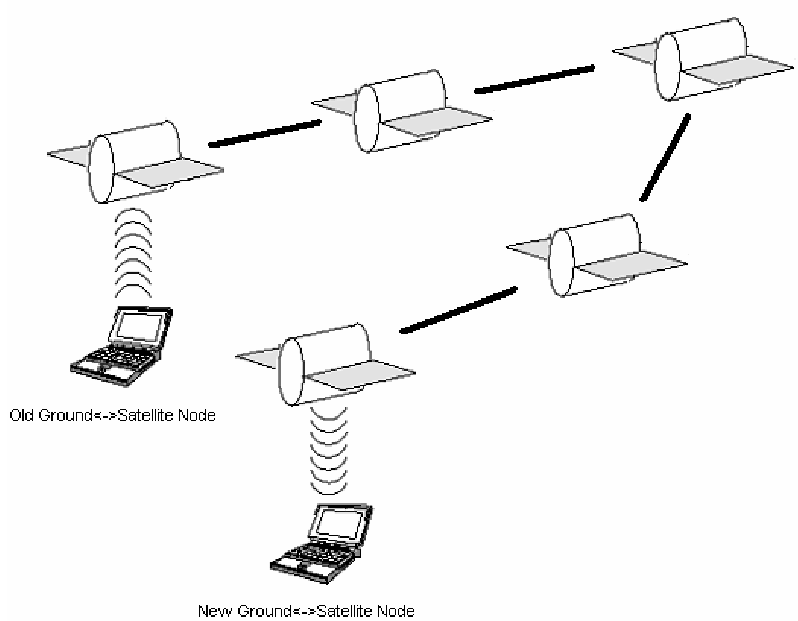


The fourth and final situation occurs when a satellite is offset from a communications link and then moves directly below the link. Alternatively, the satellite is directly below the link and becomes offset from the link. This handoff is shown in Figure 3-8 and is the most difficult to implement due to the wide array of possibilities to account for. The shortest path may not follow the old route and may not go through an adjacent neighbor.

The Assert message handling process is initiated each time there is a route change to an existing S/G connection. The Assert message is sent to the subscriber who updates the routing tables to the sender. Additionally, because a route change could affect other subscribers for the S/G pair, the Assert message is also broadcast to all other ground nodes and specifies the S/G pair that caused the assert. If a ground node receives an Assert message for a S/G pair, and the node has an active entry for the S/G, then this

ground node also sends an Assert even though it may not be strictly necessary. The broadcasted Assert message and automatic refresh are needed to ensure that S/G entries were not inadvertently removed by the initial assert. Recall that Asserts cause one link to be rebuilt and also attempt to free the old link.

Figure 3-8 Indirect Handoff to Neighbor through Upstream Hop



To mitigate the risks associated with concurrent events, a timeout of 5 seconds is used to suppress subsequent asserts for the same S/G pair. While in the timeout interval, Assert messages are ignored unless the *override* bit is set. The *override* bit is set when a ground node changes satellite neighbors – the route must be rebuilt or the ground node will be removed from the tree. There is also a timeout interval for the *override* bit, but this period is only 0.75 seconds and handles the unusual case where two ground nodes change satellite neighbors at the same or very close to the same time. Duplicate Asserts in the system cause unnecessary overhead and can lead to unpredictable results because a race condition can develop.

3.12 Model Verification and Validation

Model verification was accomplished using a systematic approach. Simulation code was compiled for the target system. Problems with syntax and illegal statements were identified and corrected. Once the models compiled correctly, the debugging cycle began.

Debugging started with the unicast routing model implementation and the ability to pass a datagram between any two endpoints and broadcast a datagram to every router. After implementing the single datagram capability, the model was extended to send a prune to the sender if there were no subscribers at the leaf nodes (nodes with no further hops). Handling prune message and pruning the network back to the source node provided the capability for a subscriber to request to join the multicast group based on the state setup from the initial broadcast and prune. A state refresh message was added to keep network conditions constant and suppress automatic re-forwarding of a pruned link. At this point, basic multicasting was in place, so the model was expanded to incorporate the idea of sources, groups, upstream hops, and subscribers. Finally, to fully implement PIM-DM, the ability to handle multiple sources, groups, and subscribers was added to the model. This is a brief overview of the major implementation milestones, but for debugging purposes, all packets were tracked and traced to verify that:

1. Sent packets followed the shortest path available. Satellite transitions cause the shortest path to be non-optimal for a small period of time while the new shortest path is recomputed.
2. Packets sent to a specific Source/Group pair reached all subscribers.

3. The aggregate number of packets sent and received at all ground nodes was tracked to identify dropped and/or missing packets. Dropped and/or missing packet numbers were identified at the end of the debugging run, allowing a manual trace to identify points where problems occurred. The cause of the packet loss was fixed and the simulation rerun to verify the results.
4. The period of a satellite is approximately 100 minutes. A 120 minute run time ensured all possible satellite configurations were covered with an extra 20 minutes of overlap to verify that transitions between satellite periods were also handled correctly.
5. A complete and successful run from a source to a destination was a good outcome, but not necessarily a success. The outcome was deemed fully successful when the run could be performed from all other sources to different destinations using the same basic set of criteria as the first run.
6. Finally, the model was deemed "correct" when the network successfully routed packets from multiple sources, having multiple groups, to multiple subscribers.

Model validation was difficult since no physical implementation of PIM-DM exists for a satellite network. An implementation based on a draft IETF specification further complicated validation since terrestrial systems have not yet been fielded based on the draft specification [ANS02]. Three elements of the model must be validated [Jai91]:

1. Assumptions,
2. Input parameter values and distributions, and
3. Output values and conclusions.

No PIM implementation exists for a LEO satellite network, and this model was based on the latest PIM-DM draft specification so no terrestrial implementation is available either. Basing the model on the draft document was chosen to take advantage of PIM-DM protocol maturation and improvements the updated specification provides.

Earlier work on similar topics [Pra99, Tho01, Fos01] and expert intuition were integral in providing necessary validation for assumptions made in the model. Timing values and basic operation of PIM-DM were extracted from the protocol specification with minimal changes to adapt to a satellite network and the idiosyncrasies that the network introduced.

Underlying network model validation was accomplished by sending packets in a unicast network. Source and destination addresses were assigned randomly and packets were sent and received at all ground nodes.

Input parameters were chosen to closely match parameters in [Tho01]. The choice of packet distributions and sizes were taken from actual network data to emulate the behavior of real systems. Ground nodes were placed at locations on each continent to provide whole earth coverage and force the network to use a wider array of routes.

3.13 Summary

This chapter presented the methodology for the experimental stage of this thesis. Parameters, factors, validation and verification of the model, and experimental design were all presented.

4. Analysis

4.1 Introduction

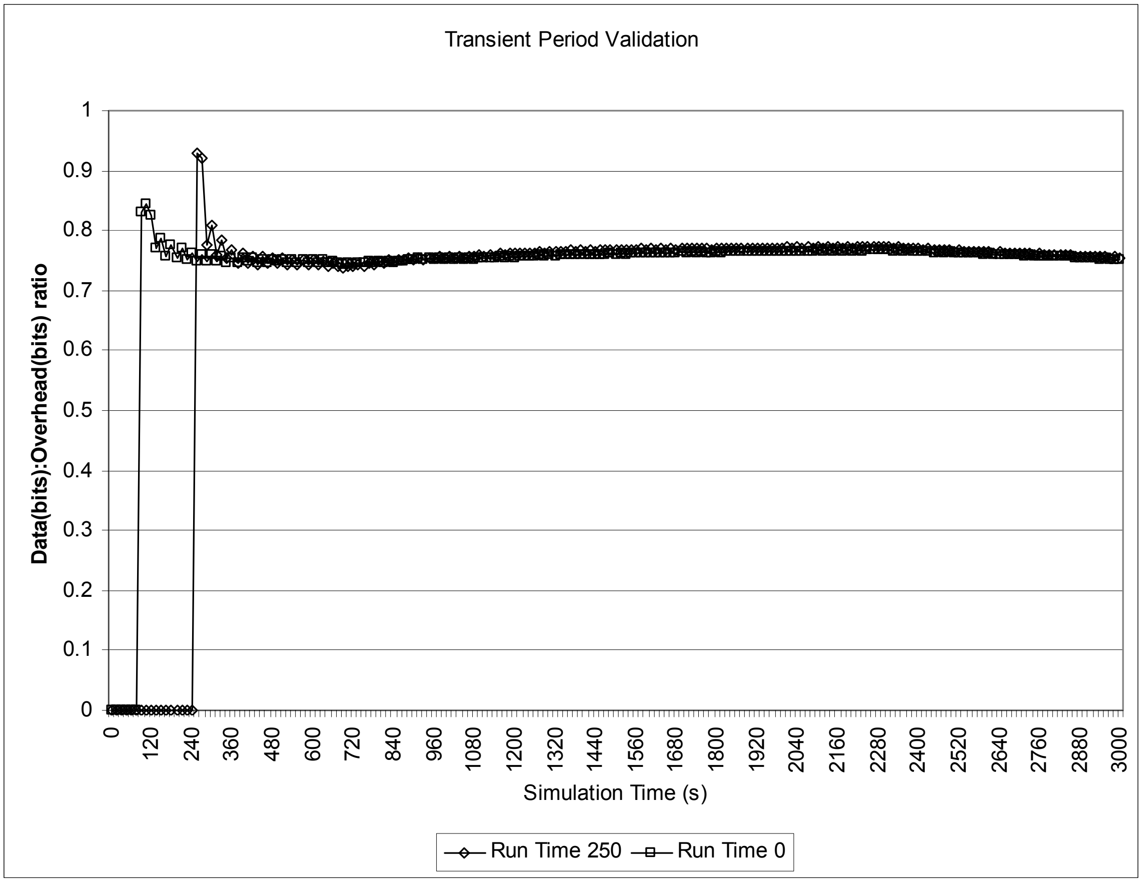
This chapter presents simulation results and analysis. Before explaining the simulation results, a brief overview of the statistical methods used is presented. Following this overview, membership levels and how these membership levels translate into sparse and dense subscribers is explained. The next section presents routing protocol results that were common across all simulations. Next, results for the PIM-DM trials are shown in two sections based on distribution methodology: One-to-Many and Many-to-Many. The conclusion of this chapter compares and contrasts PIM-DM results to prior research results for ODMRP and DVMRP.

4.2 Statistical Overview

The goal of any research is to correctly and succinctly present results in an unbiased fashion. This section explains the methods used to determine results and provides a brief overview of how statistical values are generated and applied.

Pilot studies and preliminary simulations illustrated that initial transient period data was absorbed into the statistics and did not affect the end results, as shown in Figure 4-1. The statistical results were calculated by the simulation upon receipt of each packet, so there were thousands of observations during a simulation run. The transient period was over within the first 250 seconds of simulation time. While transient period results are noticeably different than steady-state, the final result of using the samples across 0 to 250 seconds was the same.

Figure 4-1 Transient Period Validation



4.2.1 Simulation Caveats

Simulation termination times differed based on resource constraints of the computers running the simulation. The simulation allocates large amounts of dynamic memory to track and maintain the state of the multicast network. Trial runs were used to determine how long each set of simulation sequences could execute without crashing from memory allocation errors. OPNET® is provided for two platforms, Microsoft Windows and Sun Solaris 8. Both platforms experienced the same memory errors when the memory allocated exceeded approximately 2GB. This problem occurred regardless of the amount of RAM (ranging from 512MB, 1GB, and 2GB physical ram) and operating system (Windows 2000, Windows XP, and Solaris 8) running the simulation.

4.2.2 Simulation Statistics

Simulation sets are divided into 15 groups. Groups are subdivided into three distinct loading levels and each group is executed three times, with different random seeds, to achieve the desired confidence interval width. This same process was repeated twice more with different starting points yielding 405 total experiments.

4.2.3 Confidence Intervals

The confidence level chosen for this research is 90%. A 90% confidence level means that for a given mean, there is a 90% probability that the actual mean lies inside the interval [Jai91]. The confidence interval is

$$\left(\bar{x} - z_{1-\frac{\alpha}{2}} \, {}^{s}\!/\!\sqrt{n}, \bar{x} + z_{1-\frac{\alpha}{2}} \, {}^{s}\!/\!\sqrt{n}\right) \tag{4.1}$$

where $\bar{x}$ is the sample mean, $z_{1-\frac{\alpha}{2}}$ is the $(1-\frac{\alpha}{2})$ quantile of a unit normal variate, σ is the variance, s is the standard deviation, and n is the number of samples. When comparing two means, if the confidence interval contains the other mean, then the two items being compared are statistically equivalent. If the confidence interval does not contain the mean, then the items being compared may be statistically different at this confidence level.

4.2.4 Coefficient of Variation

The Coefficient of Variation (C.O.V.) [Jai91], is the ratio of standard deviation to sample mean

$$C.O.V. = \frac{s}{\bar{x}} \tag{4.2}$$

A C.O.V. of less than 10% is used as a stopping criteria for simulations, unless hardware resource requirements caused the simulation to finish earlier.

4.2.5 Analysis of Variance

ANalysis Of VAriance (ANOVA) is used to determine interactions between the primary effects, secondary effects, and tertiary effects [Jai91]. ANOVA is a method by which the variance attributable to each factor is calculated and assigned a percentage of the total variation. A single factor contributes to the primary effects, interactions between two factors are secondary effects, and finally, interaction between three factors generates the tertiary effects. In all cases, the sum of the squares for the determined effect is divided by the total sum of squares. The final step in the analysis is to perform an F-test to determine the significance of the allocation at the given significance level.

The ANOVA analysis is only valid if the assumptions below are satisfied:

1. Residuals versus predicted responses should show no trend when plotted on a scatter plot
2. Normal quantile-quantile plot should show a straight line of data points with little (or no) deviation

The method of calculating ANOVA tables is presented below for a three factor experiment [Jai91]. Equation 4.3 is the total sum of squares for all three factors. Equations 4.4 and 4.5 show primary sum of square effects for factor A and B. Equation 4.6 shows the combined sum of squares effect for factor AB.

$$SS_T = \sum_{i=1}^{a}\sum_{j=1}^{b}\sum_{k=1}^{c} y_{ijk}^2 - \frac{y_{\dots}^2}{abc} \qquad 4.3$$

$$SS_A = \frac{1}{bc}\sum_{i=1}^{a} y_{i..}^2 - \frac{y_{...}^2}{abc} \quad 4.4$$

$$SS_B = \frac{1}{ac}\sum_{j=1}^{b} y_{.j.}^2 - \frac{y_{...}^2}{abc} \quad 4.5$$

$$SS_{AB} = \frac{1}{c}\sum_{i=1}^{a}\sum_{j=1}^{b} y_{ij.}^2 - \frac{y_{...}^2}{abc} - SS_A - SS_B \quad 4.6$$

4.2.6 Random Methods

Group assignments, loading levels, and seeds were assigned following a round-robin approach. This kept the numbers sequential for ease of tracking but also ensured that each iteration of the simulation was executed with different initial conditions. The initial broadcast and prune, and subscriber actions were controlled by a random number added to a global simulation constant. By seeding the simulation runs differently, packet transmissions occur at different times for each simulation iteration.

4.3 Data to Overhead Breakdown

Statistics are divided into two areas: PIM and RIP. PIM is independent of the underlying unicast routing algorithm so it is not appropriate to incorporate the PIM and RIP statistics together. Table 4-1 shows how packet types are divided between the two areas and defines the breakdown between the data and overhead quantification of the packet contents.

Table 4-1 Packet Data/Overhead Determination

Packet Type	Data	Overhead
PIM Packet	Payload of Packet	Header of Packet
PIM Assert		Complete Packet
PIM Graft		Complete Packet
PIM Graft Acknowledgement		Complete Packet
PIM Hello		Complete Packet
PIM Join		Complete Packet
PIM Prune		Complete Packet
PIM State Refresh		Complete Packet
RIP Probe	Payload of Packet	Header of Packet
RIP Report	Payload of Packet	Header of Packet
RIP Ground		Complete Packet

The data-to-overhead calculation gives an indication of how efficiently data is transmitted through the network. The higher the ratio, the higher the amount of data as compared to overhead information. The only packet types containing data are PIM Packet, RIP Probe, and RIP Report.

The received-to-sent calculation is an indicator of packet loss, either due to an invalid route, TTL expiration, or that no next hop is available (e.g., a satellite loses communications with the ground station so ignores sending the packet). Ideally, this ratio should be close to 1.0 indicating that the majority of the packets transmitted are reaching the correct destination.

Each packet is recorded at every node that is visited along the path. Therefore, an explicit calculation of the number of hops a packet required is not done. As the packet traverses the network, packet statistics are updated at each stop from the source to the subscriber.

4.4 Group Membership Levels

The results gathered for PIM-DM are not directly comparable to results for ODMRP and DVMRP. ODMRP and DVMRP were implemented to increase the load on the system by adding additional subscriber locations around the seven localized geographic areas or scattered randomly over the earth. PIM-DM increases the load by adding more groups and due to an implementation assumption; only one ground node can be assigned to a satellite.

Ground nodes are located in seven geographic areas of the world. With seven geographic regions, all combinations of loading levels included these seven regions. Increasing load was accomplished by increasing the number of groups available for subscription and how many groups each node subscribed to.

Group subscription levels were broken down into two levels: sparse and dense. Sparse mode has few subscribers relative to the number of groups provided by the membership level. Dense mode has a large number of groups at the source and all other nodes subscribe to the full set of groups. Group membership levels of 5, 10, 15, 20, 30, and 40 were used for both sparse and dense loading levels in a one-to-many configuration. Additionally, loading levels of 5, 10, and 15 members were used for many-to-many communications. Table 4-2 shows source/subscriber combinations.

4.5 Routing

PIM-DM and PIM-SM are independent of the underlying unicast routing mechanism. It is sufficient for PIM that a packet is transmitted from Source A to Destination B. A unicast routing protocol must be in place before implementing PIM. In order to mirror prior research, the RIP-like protocol implemented in DVMRP [Tho01] is

modified to work independently and provides the framework for a unicast routing model. The routing algorithm provides a shortest path between all nodes in the network. The overhead of this protocol is contained in three packet types: Probe, Report, and Ground. Of the three, the Probe and Report packets are the most prevalent since they are common to all satellite routers. The Ground message is only applicable to a ground node and its immediate satellite neighbor.

Table 4-2 Aggregate Subscriber Tally

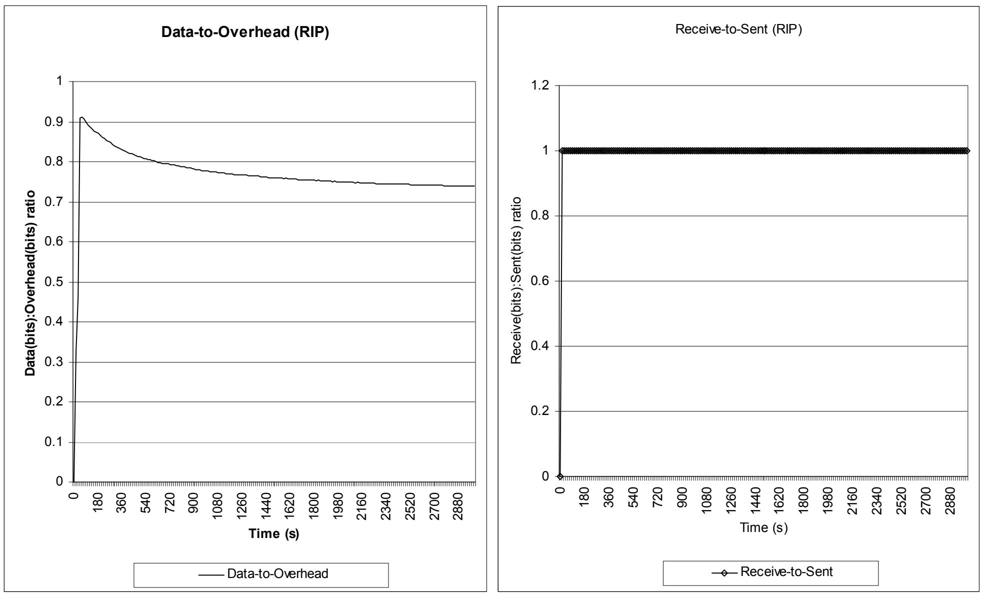
Level	Number of Sources	Number of groups per source	Number of subscribers	Number of groups per subscriber*	Total subscriber workload (users)
Low	1	1	5	1	5
Low	1	2	6	4(2), 2(1)	10
Low	1	3	6	3(3), 3(2)	15
Low	1	4	6	2(4), 4(3)	20
Low	1	5	6	5	30
Low	1	7	6	4(7), 2(6)	40
High	1	5	5	5	25
High	1	10	6	10	60
High	1	15	6	15	90
High	1	20	6	20	120
High	1	30	6	30	180
High	1	40	6	40	240
N/A	5	5	5	20	100
N/A	7	10	7	60	420
N/A	7	15	7	90	630

* The syntax *a(b)* is read as: *a* is the number of subscribers and *b* is the number of groups to which each subscriber subscribes. If a single value is present, such as *a*, then *a* groups each have *a* subscribers.

Due to its independence from PIM, the routing protocol has identical behavior for all simulation runs. There is a slight variance in the routing due to the differences in random seeds, but the actions of PIM had no bearing on the routing behavior. Routing depended solely on satellite and ground node location and because all simulations were run from time 0, routing results were almost identical. Figure 4-2 shows the Data-to-

Overhead ratio, the Receive-to-Sent ratio and the initial route discovery and setup results, for routing. The values in Figure 4-2 are almost identical to that obtained for routing in all other simulations, as RIP is also not affected by the load on the system.

Figure 4-2 RIP Statistics for All Simulations



The routing protocol propagates updates from the source of the change out to the surrounding nodes in a ring fashion every 10 seconds. This 10 second window is sufficient for updates to propagate through the network and maintain all of the links correctly. During the debug phase, an occasional routing loop would materialize due to a "hole" in the updates.

This "hole" can potentially introduce a small amount of packet loss when packets get caught in the transitory period between updates and do not reach their destination until the updated route information arrives. Essentially, the route is read from the routing tables and a short period later, updated information arrives. At this point, the packet is already being sent with both current and outdated route information.

For example, a configuration with node address 102 sending and node addresses 100,101,103,104 receiving, a loop occurred for approximately 14 seconds. This "loop" caught a portion of the packets and eventually causes the TTL to expire and destroy the packets in the loop.

Figure 4-3 Example of a Routing Loop

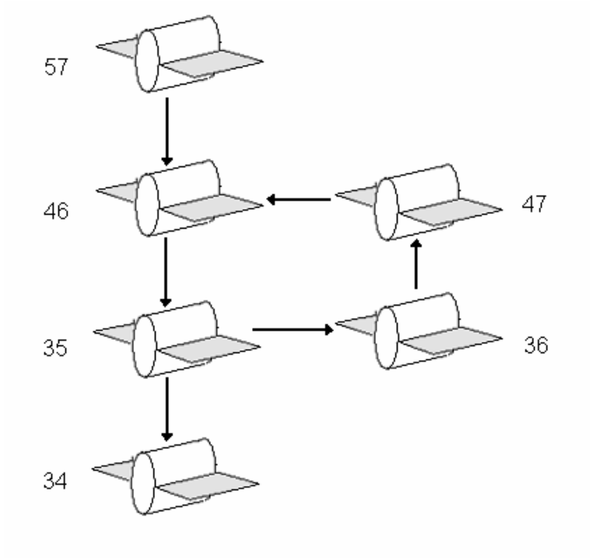


Figure 4-3 shows a packet entering the loop from 57 and loops between 46, 35, 36, and 47. The route update containing the next hop from 35 as 34 (as opposed to the outdated 36) has not yet arrived. Once current routing information arrives, the packets continue their normal journey. Barring a change to the routing protocol implementation, there is no fix for this type of behavior. It occurs occasionally and lasts for no more than 20 seconds (assuming 10 second updates between two nodes, this is the worst case).

4.6 PIM-DM Scenarios

There are two scenarios used to simulate the behavior of PIM-DM. These two scenarios have four parameters common to all experiments: density level, loading level,

membership level, and transmission type. The combination of these four parameters provides the foundation for each of the experiments.

The first scenario is built around a one-to-many multicast. This scenario is further extended to provide variations on density levels (sparse and dense) and is executed for all membership and loading levels. The second scenario is based on a many-to-many multicast and did not have a specific loading level. Instead, it used all available nodes as senders and subscribers. Additionally, this many-to-many scenario is executed for only a subset of membership levels at all loading levels.

In the one-to-many scenario, one node is chosen as the source node and the remaining nodes are subscribers. The source node generates the requisite number of groups equal to the membership and density level of the given scenario. The source node only generates enough data for the groups that have subscriptions and no additional data is generated – extra data would go through a broadcast and prune and never change from the pruned state. The broadcast and prune cycle is accomplished prior to the commencement of the subscription cycle.

The many-to-many scenario has multiple source nodes with multiple subscriber nodes. Subscriber nodes do not subscribe to their own groups, but instead subscribe to all other groups. As in the one-to-many scenario, each source generates data for the number of groups required for the membership and density level. Each source node initializes and accomplishes a broadcast and prune. Once this broadcast prune cycle is accomplished, the subscriber requests originate from all subscriber nodes. The many-to-many scenarios generate the heaviest load on the system and are the most resource

intensive. Therefore, only 5-10-15 membership levels are executed for the all-to-all scenario. Simulation time is between 300 and 1000 seconds.

The results from both scenarios are presented in the following sections. Raw data values for the simulation are in Appendix A.

4.6.1 PIM-DM One-to-Many Scenario

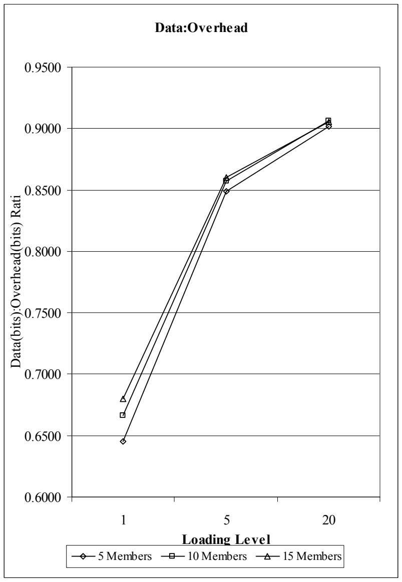
The One-to-Many scenarios are the majority of the simulation trials performed. The data for these scenarios is presented separated into Data-to-Overhead, Receive-to-Sent, and End-to-End delay.

4.6.1.1 Data-to-Overhead Analysis

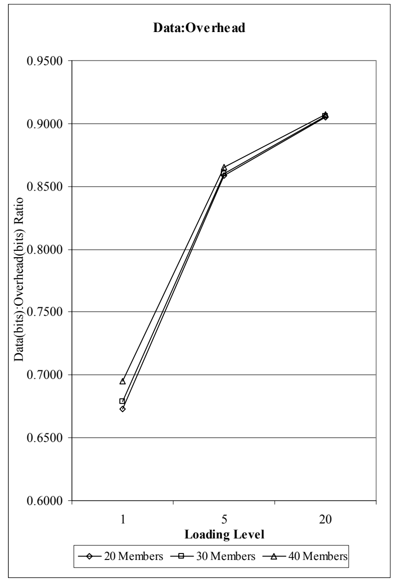
Figure 4-4 (a) (b) show Data-to-Overhead (DtO) ratio results for the Sparse 5-10-15 and Sparse 20-30-40 experiments. The experimental results for Dense 5-10-15 and Dense 20-30-40 are presented in Figure 4-4 (c) (d). Regardless of the group membership, the protocol performs similarly at each loading levels. This behavior is confirmed by ANOVA analysis (c.f., Appendix A) which finds that loading level accounts for 97% (5-10-15) and 93% (20-30-40) of the variance for each experiment respectively. Group membership accounts for less than 1% in both instances, and density accounts for less than 2%. The maximum DtO ratio is 92.2% for data packets. This is derived by dividing the mean packet size (data) by the maximum packet size (data + overhead).

The confidence intervals for the mean DtO ratio overlap for a given loading and density level. This overlap confirms that the values gathered in the simulation are all statistically equivalent for each loading level at all density levels run at that particular loading level.

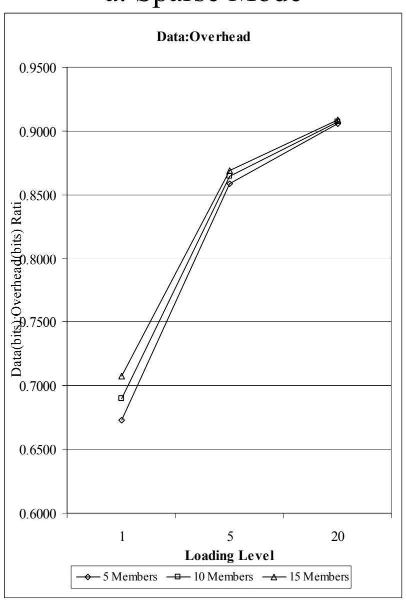
Figure 4-4 Data-to-Overhead Ratio



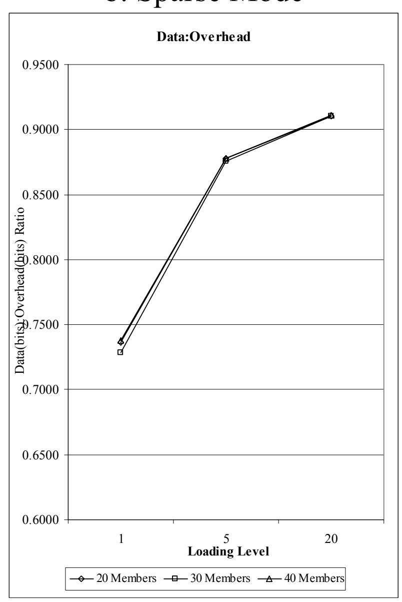
a. Sparse Mode



b. Sparse Mode



c. Dense Mode

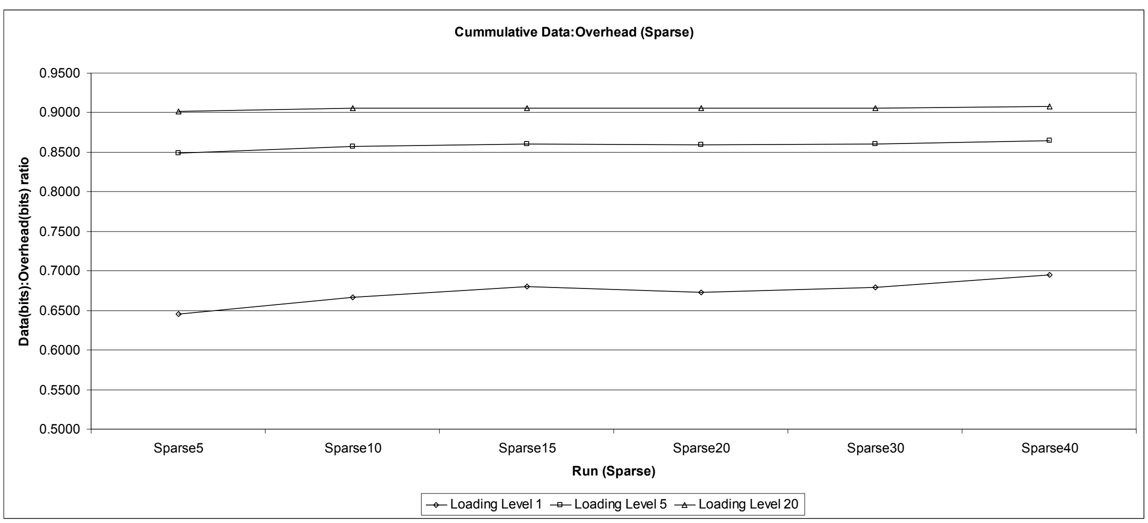


d. Dense Mode

The stability of the DtO ratio is evident when data for all trials are combined. Figure 4-5 presents all of the Sparse values contributing to the DtO ratio for loading levels 1, 5, and 20 and Figure 4-6 presents the values for the Dense trials at these same loading levels. The DtO ratio is relatively flat for all loading levels across the range of

group membership levels. As group membership increases, the loading level trends up, but this upward trend is slight as confirmed by the standard deviation and the minimum and maximum values. From Appendix A, the standard deviation away from the Sparse mean of 0.8584 is less than 0.017 and from the Dense mean of 0.8785 is less than 0.027. This upward trend is expected because group membership is increasing. The increase in group membership leads to an increase in overhead as well as data packets transferred – but the increase in data packets is greater than the overhead increase, so the overall DtO is more efficient. This same trend is exhibited by the slightly higher DtO of the Dense runs compared to the Sparse runs – the increase in group membership leads to an increase in the DtO ratio.

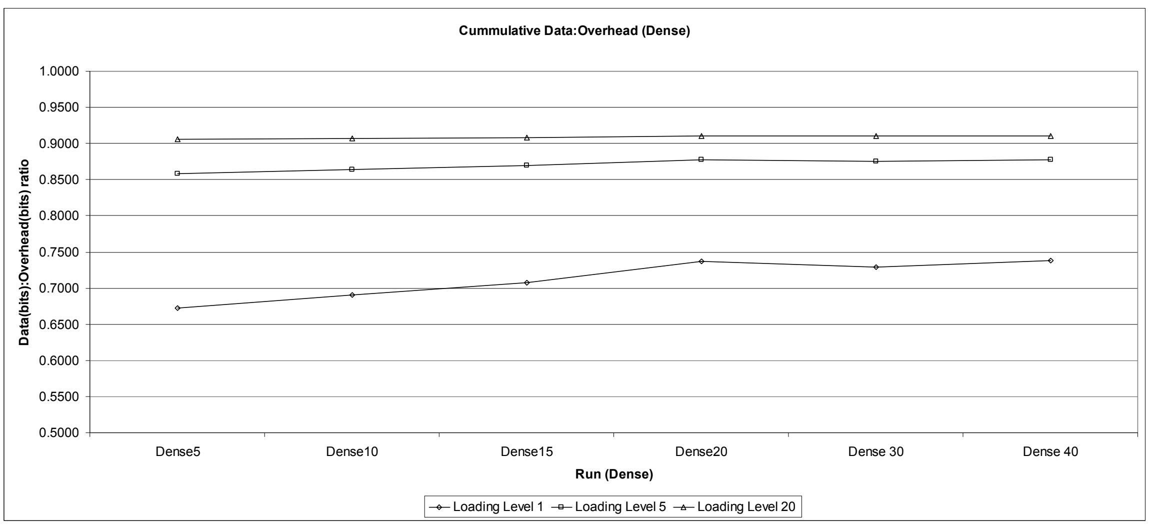
Figure 4-5 Cumulative DtO for Sparse Runs



These results for the DtO are expected; PIM-DM effectively sets up a switched network between the source and subscribers. A multicast tree is built after a Join request is received, but the state of each node is maintained through State Refresh messages. Barring changes to the subscriber list, satellites moving in their orbital planes are the only

dynamic aspect of the tree structure. As such, the overhead to maintain the tree once the multicast tree is built is the overhead necessary to account for satellite movement. The minimum and maximum values (c.f., Appendix A) closely bound the mean value with a standard deviation that is less than 4% of the mean. For non-scaled data (e.g., loading level of 1), the minimum DtO is 0.645 (Sparse) and 0.673 (Dense) and occurs at the group membership level of 5. The maximum DtO is 0.695 (Sparse) and 0.738 (Dense) and occurs at a membership level of 40.

Figure 4-6 Cumulative DtO fro Dense Runs



Different behavior is exhibited by sparse and dense mode due to the differing configuration for group assignments. Spare mode has subscriber entries evenly distributed between available nodes to equal the maximum subscriber level. Dense mode has all nodes subscribing to the maximum amount of groups available for the subscription level. The network overhead is higher in dense mode to set up trees to each

subscriber, but the significantly higher data packet load offsets the additional overhead and leads to a higher DtO ratio for dense mode.

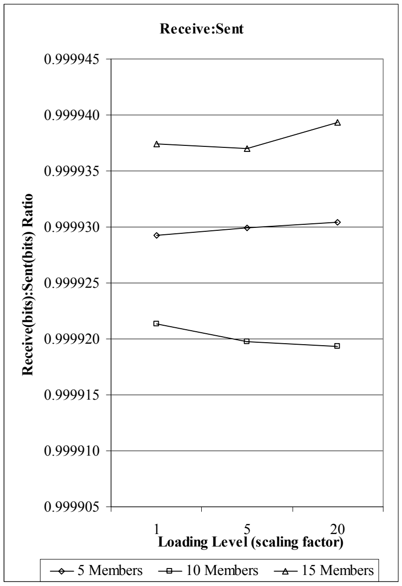
Loading level 1 is not scaled, while both loading level 5 and loading level 20 are scaled values. The scaled loading levels, as shown over the range of experiments, are indicative of the type of savings that would be realized if the network was less dynamic. Potential network efficiencies are illustrated in Figure 4-5 and Figure 4-6. As the loading level increases, the DtO ratio also increases, which is expected since more packets are flowing for the same amount of overhead. Changes to the multicast tree are due to satellite movement so a loading level of 5 (or 20) sends 5 (or 20) times the aggregate number of data packets as a loading level of 1. Thus, considerably more data is flowing through the network for a static amount of overhead, leading to a favorable increase in the DtO ratio.

Lastly, the DtO ratio for all experimental runs shows that PIM-DM scales well with load. There is no statistical difference between the various density levels at a given loading level. Therefore, the protocol behaves almost identically in all instances. Scalability is crucial when the load in the system varies and the protocol should ideally behave similarly across loading levels.

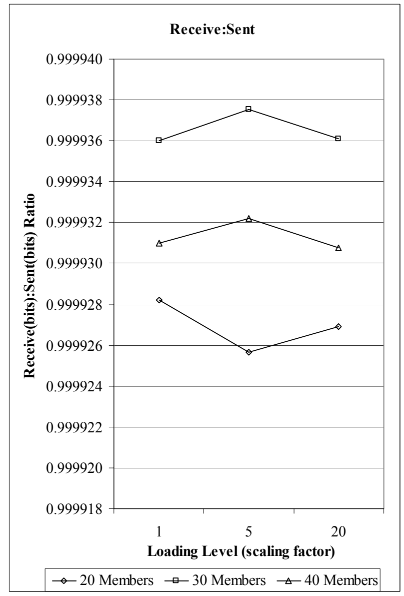
4.6.1.2 Receive-to-Sent Analysis

Figure 4-7 (a) (b) show the Receive-to-Sent (RtS) ratio results for the Sparse 5-10-15 and 20-30-40 runs. The Dense 5-10-15 run is shown in Figure 4-7 (c) followed by the Dense 20-30-40 run in Figure 4-7 (d).

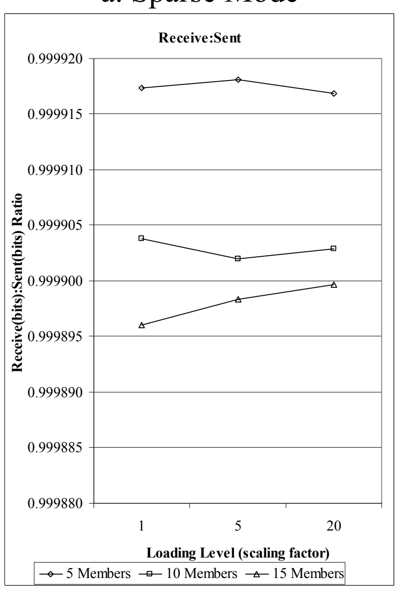
Figure 4-7 Receive-to-Sent Ratio



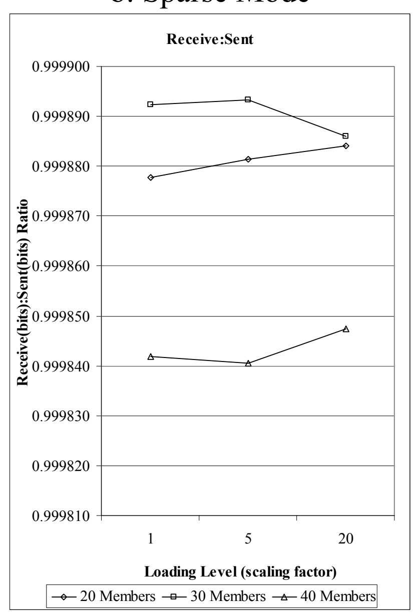
a. Sparse Mode



b. Sparse Mode



c. Dense Mode



d. Dense Mode

The system configuration presents no possibility of packet collisions for packets generated from the ground nodes. There is only a single ground station associated with a satellite, so at most one ground node is transmitting. The ground node might transmit to

two satellites during a handoff, but the single ground node per station requirement is maintained. A ground node provides a range of groups available for subscriptions, but all groups are originating with the same source. Since a collision is not possible in the network configuration being modeled, the reason for RtS being less than 100% is packet loss due to dropped packets.

As shown by Figure 4-7 (a-d), the RtS ratio appears to differ considerably, but this difference is negligible based upon the scale of the axis. The RtS is above 99.98% in all cases; loading and membership levels have no significant contribution. The ANOVA shows that the difference between Sparse and Dense mode contributes approximately 22.5% (5-10-15) and 53% (20-30-40) of the variation. The Sparse and Dense mode joined with the membership level account for 32.5% of the variance (5-10-15%) and 66% of the variance (20-30-40) by combining first order and second order effects for all loading levels. The unexplained variance for both Sparse and Dense mode is 67% and 34% respectively, and it is suspected that the location of the sending node heavily influences network behavior. The unexplained Sparse mode variance is higher (as expected) than Dense mode unexplained variance because of the inclusion of the Sparse-5 experiment which only uses 5 of 7 possible earth nodes as subscribers; all other experiments use a combination of all 7 nodes.

Like the DtO analysis, the RtS analysis shows that values obtained for the experiments as statistically equivalent. Unlike the DtO analysis, this overlap in the confidence intervals for the mean RtS value extends across all membership and loading levels.

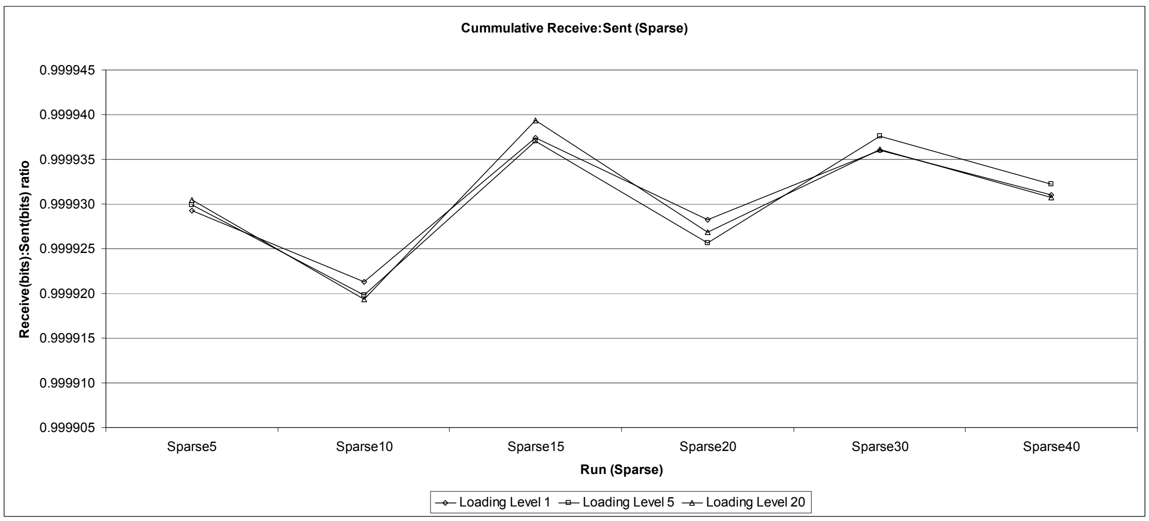
Furthermore, the RtS clearly shows that virtually all packets sent have been delivered successfully. Appendix A shows that packet loss is negligible and accounts for less than 0.02% of the packets sent for all membership levels. Using the data from Appendix A, an increase in the loading from Sparse 5 to Dense 40 illustrates a small increase in packet loss (difference in ratios is 0.0087%), but loading level does not affect the ability of the system to transmit packets successfully from the source to the subscribers. Furthermore, the dynamics of the satellite network are working properly as handoffs and reconnections are dropping minimal packets under any of the executed configurations.

For Sparse membership levels, Figure 4-8 exhibits a saw-tooth pattern that increases with Membership level. The change in RtS values is relatively small (approximately 99.992% to 99.994%) and is statistically insignificant, but each sample point is grouped closely in the same cluster. Clustering of the data points is not related to the starting point for the multicast tree because three separate runs were executed from three different starting points. Instead, the saw-tooth pattern is attributable to the method used in assigning the membership levels for the experiment. The unbalanced assignments happen at the 10, 20, and 40 membership levels – where the groups are split in a 66.7% and 33.3% configuration.

The RtS graphs as shown in Figure 4-8 and Figure 4-9 are visually different from each other, unlike the behavior exhibited by the DtO graphs. While the values obtained are statistically insignificant, the charts seem to indicate a general trend for both sets of density levels. This trend is explained by the Sparse runs ranging from a total of 5 subscribers to 40, and the Dense experiments ranged in size from 25 to 240 users. The

Dense runs have a much higher subscription level than the Sparse runs, and this higher subscription level corresponds to a higher data packet level, leading to more packets in transition in the network. The additional packets of the Dense runs provide more opportunity for a packet to get discarded so the RtS level trends down slightly.

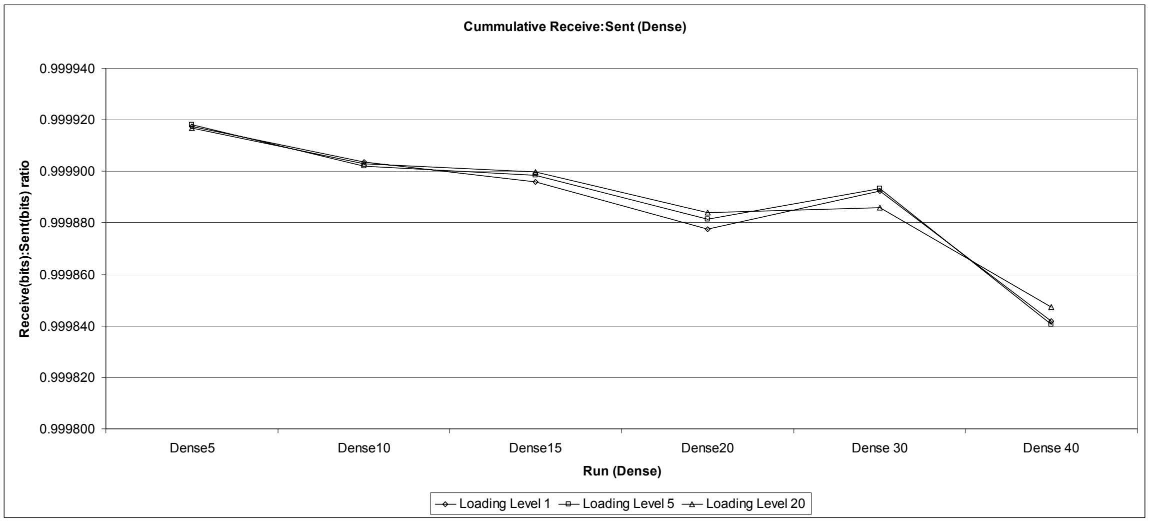
Figure 4-8 Cumulative RtS for Sparse Runs



The RtS level for the Dense experiments, as shown in Figure 4-9, steadily decreases with an increase in membership level, with the exception of Dense 30. The data is initially very concentrated, but as membership level increases, the data becomes more dispersed. This behavior is expected with the increase in the number of groups subscribing to the source. Group sizes are increasing in tandem with increase in membership level causing a multiplicative increase in the number of packets sent. For instance, Figure 4-9 shows that Dense-10 has 6 nodes subscribing to 10 groups each, and one source generating data for 60 groups. The Dense-40 has 6 nodes subscribing to 40 groups each, with one source node generating data for 240 groups. The median number of packets flowing during a given time interval is 60 for the Dense-10, but is 240 for the

Dense-40, a 4-fold increase in transmitted packets. The increased packet load is results in a slightly lower RtS ratio as the probability increases that a packet may be dropped.

Figure 4-9 Cumulative RtS for Dense Runs

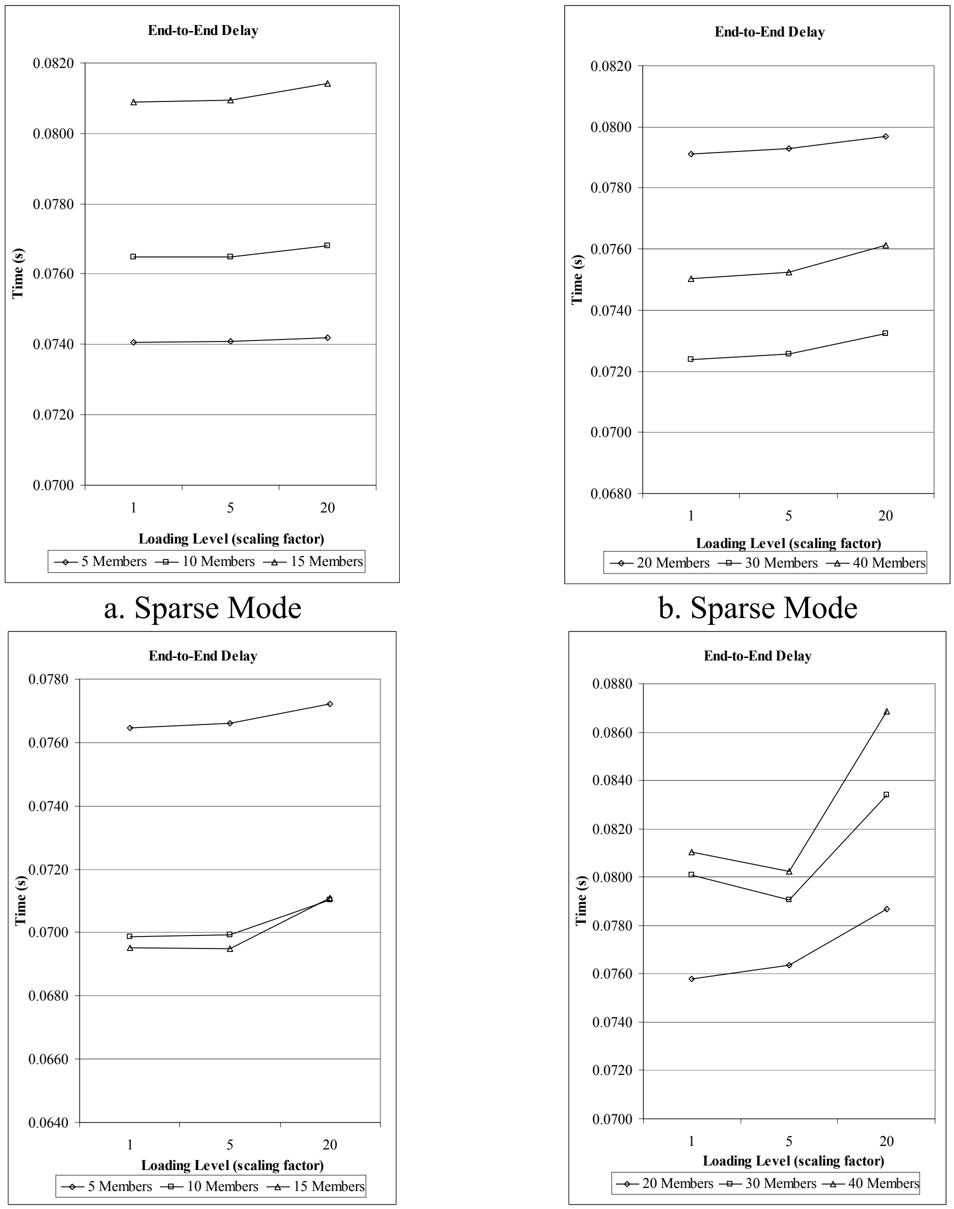


4.6.1.3 End-to-End Delay Analysis

The End-to-End (EtE) delay exhibits greater variability based on loading level and group membership levels. Figure 4-10(a)(b) present the Sparse values, while Figure 4-10(c)(d) show the Dense values. The linear increase that would be expected for EtE as loading level increases is not present.

Dense group membership levels were grouped closer together than the Sparse group membership levels, but are stable across all loading levels. The Sparse group membership level data tracked closely across loading levels, but exhibited more variability in EtE delay times, whereas the Dense group membership was also closely grouped across all loading levels but did not include the variability of the Sparse experiments.

Figure 4-10 End-to-End Delay



a. Sparse Mode

b. Sparse Mode

c. Dense Mode

d. Dense Mode

As in the RtS analysis, the confidence intervals for the mean EtE time overlap across all loading and membership levels. This result is unexpected as there appears to be a significant difference in the Sparse density levels. The one-to-many runs have approximately 25% of their EtE variance accounted for by density and membership

effects. Furthermore, the 5-10-15 experiments have an unexplained error rate of 74% while the 20-30-40 experiments have a rate of 70%. As was the case for the RtS analysis, the unexplained variance clearly shows there are other factors affecting the behavior of the network. It is suspected that the EtE, while influenced by the location of the subscriber nodes, is also affected by other factors such as aggregate number of hops, distance between satellites, transmission distance between the ground node and the satellite neighbor, and the number of satellite to ground node transitions (forwarding nodes increase the number of hops, but limit packet loss).

Cumulative EtE values for Sparse membership levels are shown in Figure 4-11 and the Dense membership levels are shown in Figure 4-12 trends slightly down for Dense 10 and 15 but trends upward for the other loading levels. Due to the small difference in values, this is attributable to the insignificance of the mean values and is statistically not a factor. In general, the Dense runs are performing as expected and exhibiting a higher EtE time for higher loading levels as queuing occurs more often.

The EtE was expected to increase as the load on the system increases. There is no consistent trend exhibited in either Figure 4-11 or Figure 4-12. Trends are slightly down for Dense 10 and 15 but upward for the other loading levels. Due to the small difference in values, this is attributable to the insignificance of the mean values and is statistically not a factor. In general, the Dense runs are performing as expected and exhibiting a higher EtE time for higher loading levels as queuing becomes more common.

As Figure 4-12 illustrates, the EtE trends down for Dense 10 and 15 but trends upward for the other loading levels. Due to the small difference in values this is attributable to mean value insignificance and is statistically not a factor. The Sparse runs,

which again are statistically equivalent, present much different trend line behavior as shown in Figure 4-11. Though the variations in EtE delay may seem counter-intuitive, the range of delay value differences is less than 10 ms. This delay time roughly equates to the time it takes a signal to propagate through a single link at a low elevation angle or through a cross-link for satellites close to the Equator. As more members are presented, the distribution of ranges decreases as partial clustering can take place. This causes the number of hops a packet must travel to slightly decrease, resulting in slightly lower delay values even though the number of members has increased.

Figure 4-11 Cumulative EtE for Sparse Runs

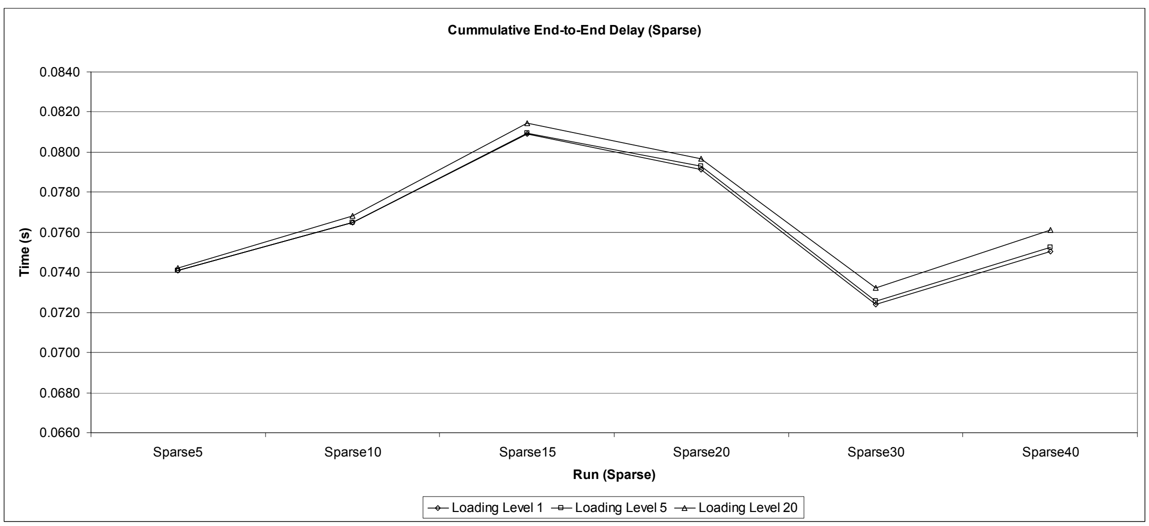
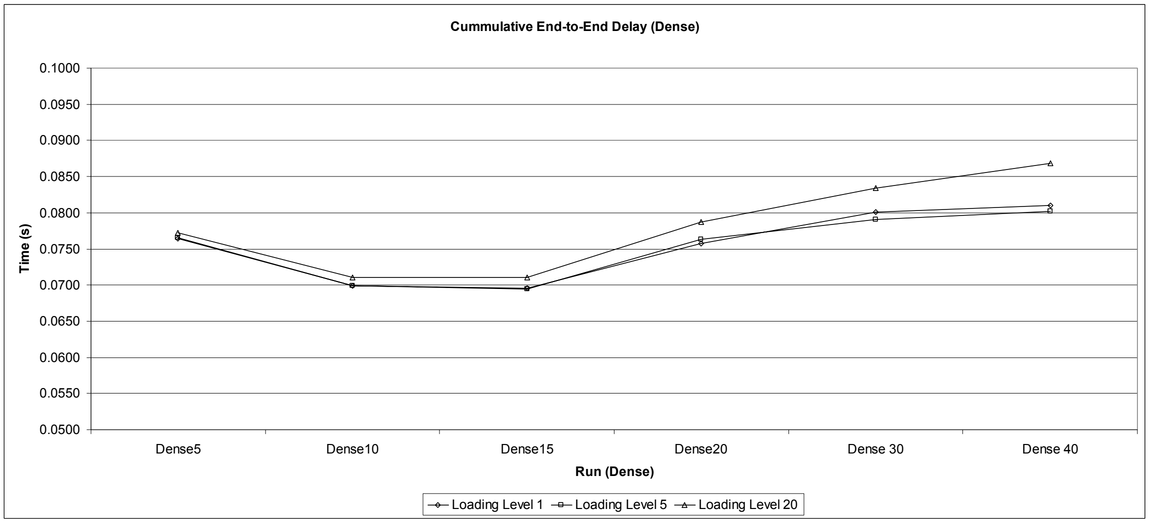


Figure 4-12 trends slightly down for Dense 10 and 15 but trends upward for the other loading levels. Due to the small difference in values this is attributable to the insignificance of the mean values and is statistically not a factor. In general, the Dense runs are performing as expected and exhibiting a higher EtE time for higher loading levels as queuing occurs more often.

Figure 4-12 Cumulative EtE for Dense Runs



The similarity in the results is also shown in Appendix A, as the mean value achieved across loading and density levels ranges between 0.0753 and 0.0780. While the ETE trends slightly higher with the increase in the loading level, the ISL links are able to handle the greater packet size and transmit the data in approximately the same time. The mean as shown in Appendix A of loading level of 5 is 0.0764 and 0.0753, and only 0.0769 and 0.0780 for loading level 20 – or a very minor impact to the EtE delay for the additional data from the scaling metric.

4.6.2 PIM-DM Many-to-Many Scenario

The many-to-many experiments are used to determine how the PIM-DM implementation performs under heavy load. In the many-to-many scenarios, every ground node generated the requisite number groups to meet the membership level. Every other ground node subscribed to all of the groups at all of the other nodes (excluding itself). For example, the All-10 run would have 7 sources providing 10 groups, and 7 subscribers to 60 groups each.

The simulation end times ranged in value from 1000 seconds for the All-5 run to only 300 seconds for the All-15 run. The all-to-all runs were extremely resource intensive and the simulations would not run to completion in all instances due to memory being exhausted or OPNET® running out of event handles.

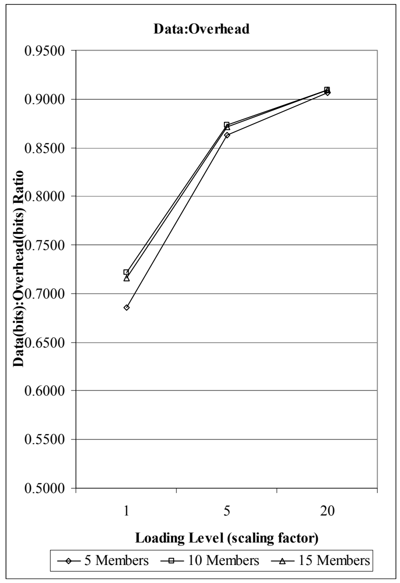
The ANOVA tables for the Many-to-Many scenarios use the average values of the cumulative statistics. Due to the inability to get a complete set of runs for All-15, the number of data points is not consistent. The inconsistency in the number of data points precluded the use of the standard ANOVA calculations which assume that there is a consistent number of data points for all runs.

4.6.2.1 Data-to-Overhead Analysis

The results for the Data-to-Overhead (DtO) ratio are in Figure 4-13. As was shown in the One-to-Many DtO analysis, the simulation values are grouped closely together at the differing loading level points. The simulation is performing similarly at each loading level regardless of group membership. The ANOVA confirms this observation as 98% of the variance is attributed to workload. Both group membership and density account for less than 1% each.

The confidence interval results are different from the One-to-Many DtO results. First, the results obtained for the All-to-All scenario are statistically significant when compared with the One-to-Many Sparse configuration. The confidence intervals do not overlap for any loading or membership levels. Conversely, when the All-to-All scenarios are compared to the Dense One-to-Many configurations, the results are statistically identical. The results are statistically significant for the loading level as those confidence intervals do not overlap.

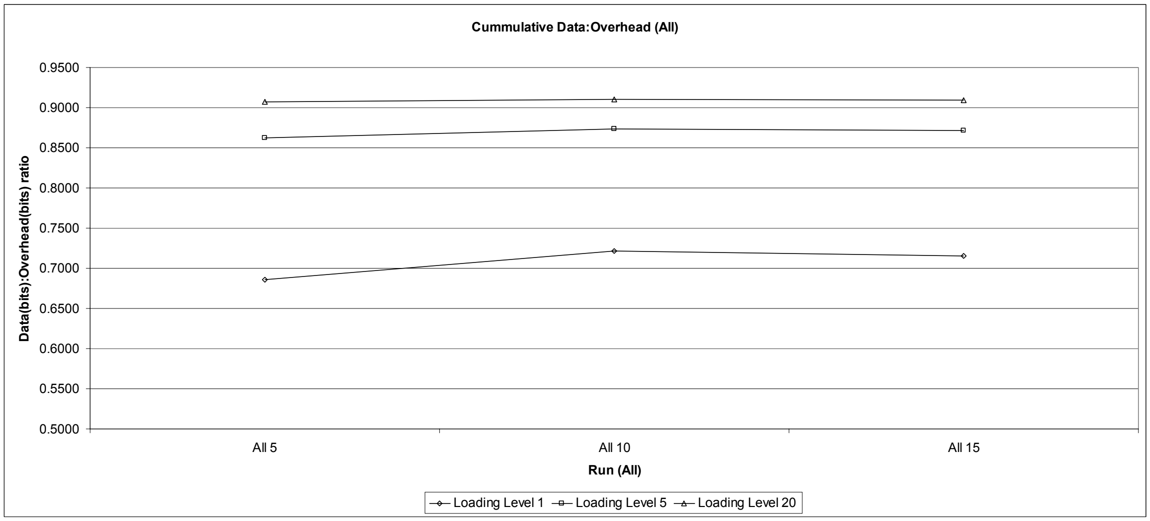
Figure 4-13 Data-to-Overhead Ratio for All-All Runs



These results are expected. The All-to-All runs are similar to the One-to-Many Dense runs, with the addition of more subscribers and consequently more groups. The additional overhead to setup all of the subscriber networks is offset by the increased data packet load, leading to a DtO ratio very similar to the One-to-Many Dense runs. This observation is further confirmed by comparing the mean values of Appendix A, the means are close in magnitude and as was shown previously, statistically equivalent.

Figure 4-14 presents the cumulative EtE statistic across membership and loading levels. The results are statistically equivalent for all membership levels at a given loading level. This is also visible by the relatively linear nature of the line at each loading level – little variation is visible with the increased loading caused by additional members.

Figure 4-14 Cumulative DtO for All-All Runs



Even under the intense loading of the All-to-All case, the PIM-DM protocol yields similar results to the lightly loaded case. This adds credence to the idea that PIM-DM scales well with additional load with little impact on the DtO ratio.

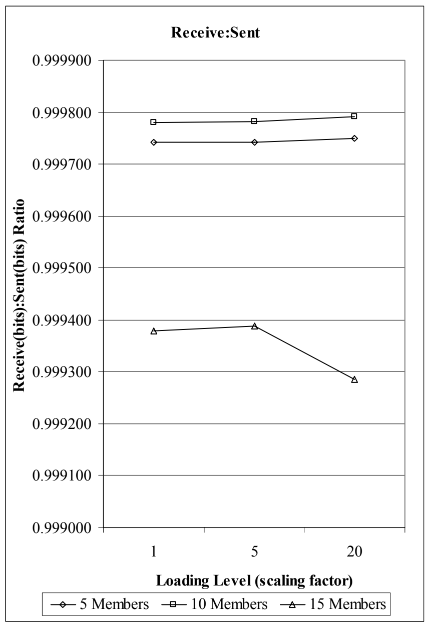
4.6.2.2 Receive-to-Sent Analysis

Figure 4-15 shows the Receive-to-Sent (RtS) ratio for the All-to-All 5-10-15 runs. There is still no possibility of collisions as the system has only a single satellite assigned to each ground node regardless of the number of groups for which the ground node is generating data. All ground nodes provide a range of groups available for subscription to attain the required membership level. As before, since collisions are not possible the only reason that the RtS is not 100% is due to packet loss.

According to Figure 4-15, the RtS ratio appear to differ considerably, but this difference is again negligible based upon the scale of the axis. RtS is above 99.92% in all cases but the differences in loading levels are more pronounced. This difference in trends is due to the inability to execute a complete set of simulation runs for the All-15 runs. For the three simulation executions, only one sequence was able to complete all the 15

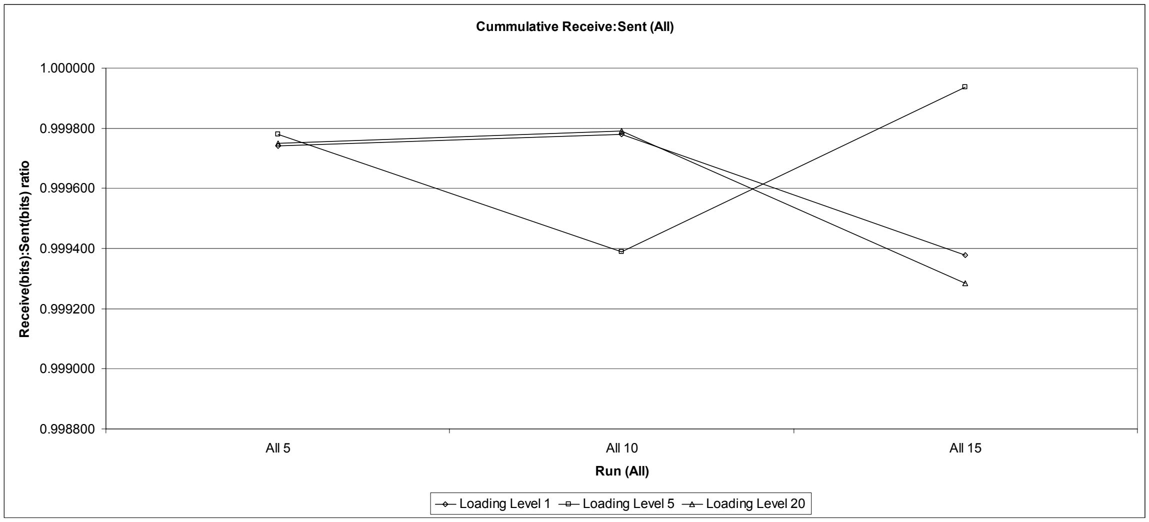
runs, the second sequence was able to complete 66% of the 15 runs, and the third sequence could not complete any of the 15 runs. This inability to execute a complete execution was due to a variety of factors, but appeared to be related to the simulation code and how OPNET® handled the event and memory allocations.

Figure 4-15 Receive-to-Sent Ratio All-All Runs



The ANOVA analysis shows that density level and membership level account for approximately 99% of the variation. The RtS values for the All-to-All runs are statistically significant when compared to both of the One-to-Many density levels as the confidence intervals never overlap. Confidence intervals do overlap inside the All-to-All trial, therefore, the RtS executions are statistically equivalent across all loading, membership, and density levels. This result was expected as the differences in means is relatively minor and only highlighted by the scale of Figure 4-16.

Figure 4-16 Cumulative RtS for All-All Runs

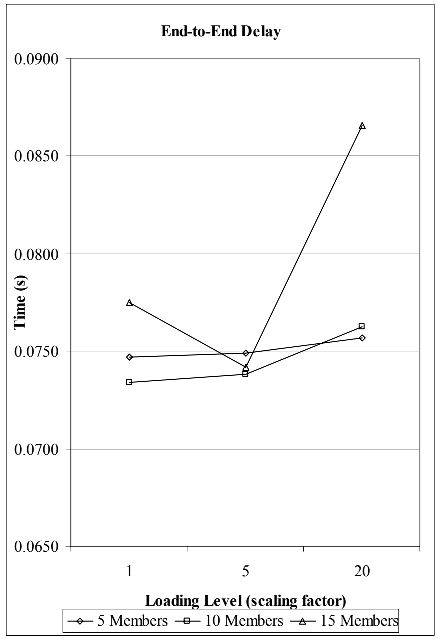


As was the case in the one-to-many scenarios, the high RtS ratio means that few packets are dropped and the majority are delivered to the destination successfully. Appendix A provides a complete breakdown of RtS levels, and in the worst case 0.08% of packets are unsuccessfully delivered. Excluding the incomplete All-15 runs, approximately 0.025% to 0.065% of packets are dropped. Even with the higher loading introduced by this series of simulation runs, the protocol as implemented has a high success rate at delivering packets.

4.6.2.3 End-to-End Delay Analysis

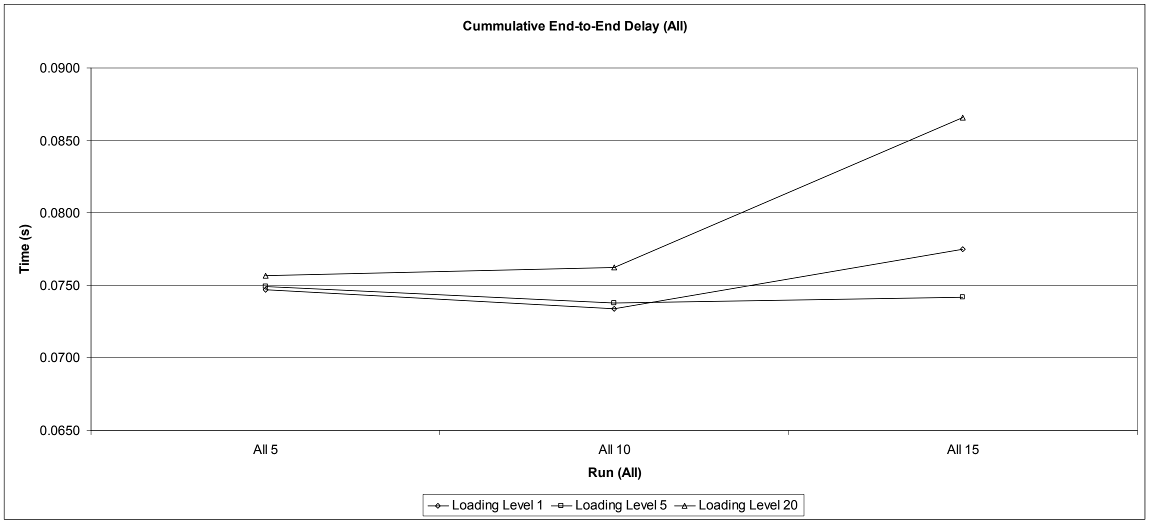
The End-to-End (EtE) delay exhibited large variability based on loading level and group membership levels and is shown in Figure 4-17. The expected response would be a linear increase as loading increases, but again the data does not support that.

Figure 4-17 End-to-End Delay for All-All Runs



The ANOVA analysis does not explain the variance, there is no predominant factor, and the variance is spread between the first, second, and third order effects. The large variability in the EtE is shown clearly in Figure 4-17. This variability is especially pronounced for the All-15 run. Additionally, the confidence intervals for the EtE overlap at all membership and loading levels, so the results are statistically insignificant. This result is not as obvious from Appendix A, which indicates that there is a greater difference between the means of the EtE delays. The range of variability for the EtE values represents less one round trip time for signal propagation time. While Figure 4-17 seems to imply a dramatic variation, this variation can be explained through paths that can vary in length by two to three hops. These variations in hop distances are not uncommon given the geographic separation of the earth stations and the possible forwarding which can occur during ground to satellite transitions.

Figure 4-18 Cumulative EtE for All-All Runs



The lack of additional data points for the All-15 trial make it impossible to draw firm conclusions. From Appendix A, the All-5 trial has EtE values grouped closely, while the All-10 showed a decrease in EtE for two of the loading levels and greater dispersion between the data points. Ignoring the incomplete data of All-15, the EtE does appear to be similar regardless of loading, density, and membership level.

4.6.3 Protocol Comparison

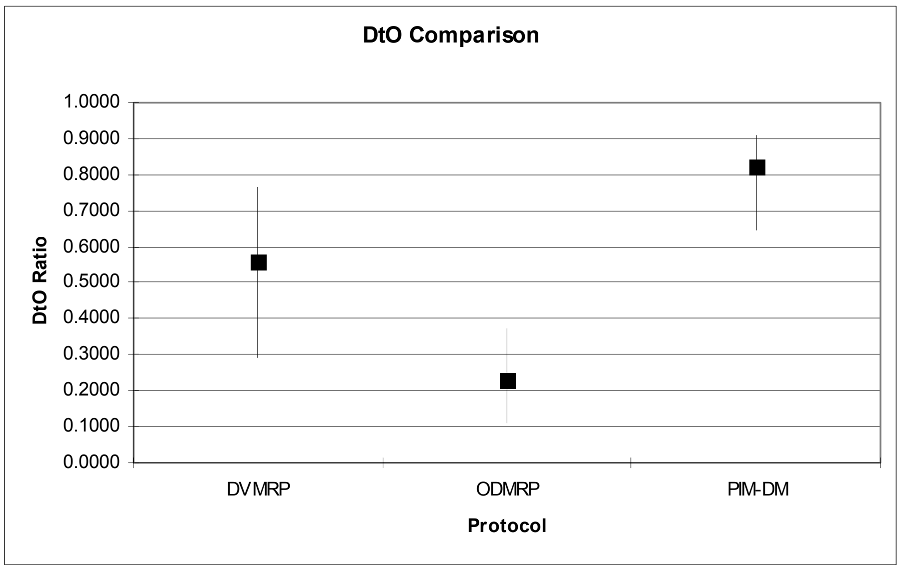
The final task is to compare the results from the PIM-DM runs to the results gathered earlier for ODMRP and DVMRP. This comparison is not possible in a strictly one-to-one fashion since the protocols differed considerably in their implementation and execution. This comparison can be accomplished by examining the ranges and bounds of all three protocols at comparable loading levels.

A summary of the raw data values for ODMPR, DVMRP, and PIM-DM is included in Appendix B. The data being compared spans all membership and density levels for each protocol, but does not include the data related to satellite failures. For a complete description of ODMRP and DVMRP results refer to [Tho01].

4.6.3.1 Data-to-Overhead

The Data-to-Overhead (DtO) ratio varied considerably between all three protocol implementations. A summary of the pertinent data (low, high, and average value) is shown in Figure 4-19.

Figure 4-19 DtO Comparison

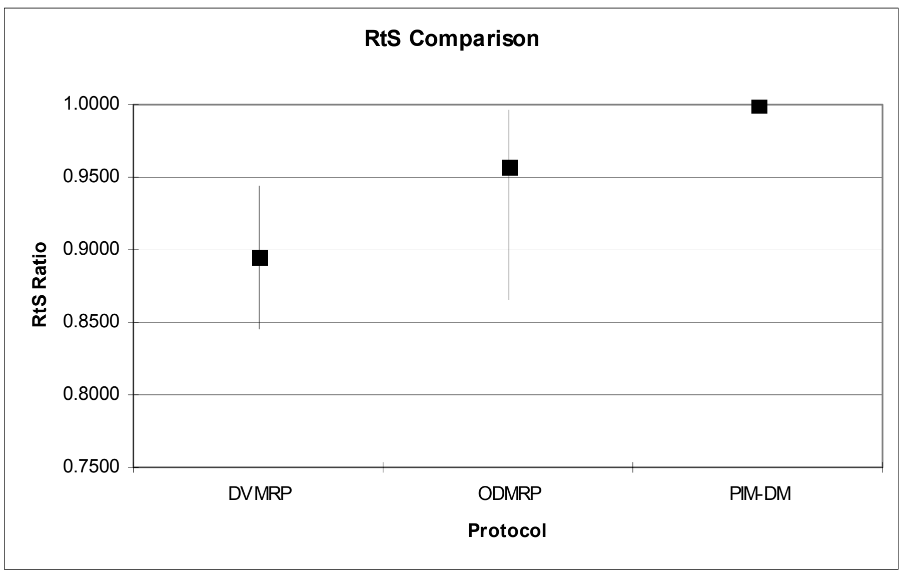


As is shown by Figure 4-19, the DtO measurements illustrate that PIM-DM has a consistently higher DtO ratio than the other two protocols. This is exactly the result that is expected and is caused by the separation of the routing protocol from the multicasting protocol. Both DVMRP and ODMRP had the routing protocol embedded in the multicasting packet, so there was a dependency between routing and packet data. Because the routing protocol and PIM-DM are independent, actions taken by PIM do not cause a corresponding reaction in RIP. But excluding the routing protocol causes PIM-DM's 82% average DtO ratio to be much higher than both DVMRP (56%) and ODMRP (23%) combined. The benefit of separating the routing protocol from the multicasting protocol is clear as the DtO ratio increases.

4.6.3.2 Receive-to-Sent

The Receive-to-Sent (RtS) ratio is consistently higher for PIM-DM than for either ODMRP or DVMRP as shown in Figure 4-20. The near perfect transmission capability exhibited by PIM-DM is ideal when data integrity is crucial. A higher RtS not only reliably delivers packets, but decreases the necessity to retransmit missed packets. The average RtS for PIM-DM was 99.98% as compared to DVMRP's 89.51% and ODMRP's 95.66%. The range for both DVMRP and ODMRP was also much wider; DVMRP ranged from 84.5% to 94.4%; and ODMRP ranged from 86.6% to 99.7%.

Figure 4-20 RtS Comparison



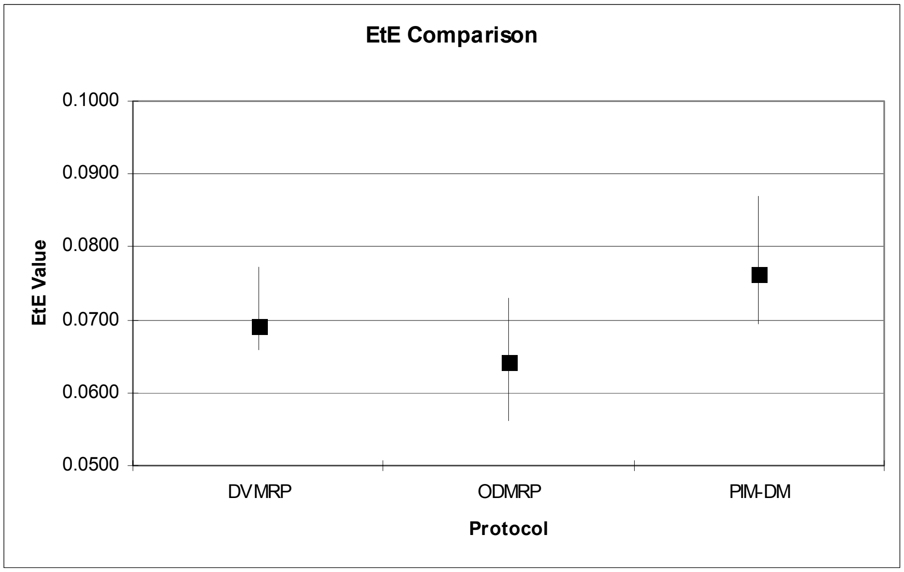
PIM-DM had a smaller range for RtS (99.93%-99.99%), regardless of the loading on the network. The PIM-DM protocol has a much greater transmission reliability ratio and therefore is a better choice than either DVMRP or ODMRP if RtS is the sole factor.

4.6.3.3 End-to-End Delay

The End-to-End (EtE) delay factor was lower for both ODMRP and DVMRP when compared to PIM-DM. As shown in Figure 4-21, both ODMRP and DVMRP have

an average EtE of under 0.07 seconds, and PIM-DM has an average EtE of 0.076 seconds.

Figure 4-21 EtE Comparison



This result is surprising, especially when considering that both DVMRP and PIM-DM should have built approximately the same length tree structure. The difference can be explained by two factors. The additional layer between PIM and RIP introduces a slight delay. As with a variety of other protocol implementations, the layers of the protocol stack introduce additional latency. In this case, the packet has to be "sent" by PIM-DM and the routing protocol had to query the routing tables to determine the next hop before actually sending the packet. Secondly, the 99.98% RtS achieved by PIM-DM has the price of non-optimal paths while the connection between the source and the subscriber is being renegotiated. To avoid dropping packets, the old satellite node would forward packets to the new satellite node until the connection was removed. This forwarding capability increases reliability but introduces additional hops along the path leading to higher EtE delays.

4.7 Conclusion

PIM-DM provides a scalable framework for a LEO satellite communications environment. The protocol scales with load and provides equivalent performance characteristics regardless of the system load. The Data-to-Overhead ratio is on average 80% and increases with a more stable network configuration. The Receive-to-Sent ratio is 99.98% across all loading levels, so few packets are dropped and the majority are delivered successfully. Finally, the End-to-End delay of PIM-DM is approximately 76 ms which is satisfactory for packet based network communications.

PIM-DM compares favorably to both DVMRP and ODMRP and in some cases surpasses the performance of both protocols. The separation of routing and multicast data simplifies the network and allows easier migration to other multicast protocols. PIM-DM gives the user superior transmission capability and provides a reliable, scalable network configuration with little packet loss and excellent responsiveness.

5. Conclusions

5.1 Restatement of Research Goal

Satellite multicasting is the focus of various research efforts [Pra99, Tho01, Fos01]. The focus of this research is to extend prior work by incorporating Protocol Independent Multicasting-Dense Mode (PIM-DM) into a model defined for ODMRP and DVMRP [Tho01]. Prior research focused on the overall network system performance of a six plane, 66 satellite LEO constellation using either ODMRP or DVMRP. This model is extended here to include the PIM-DM protocol.

5.2 Research Contribution

This research is the first to implement and analyze PIM-DM behavior in a LEO satellite network environment. This work also introduces a simple unicast routing algorithm to transmit packets for the PIM protocol. While this unicast routing algorithm is not unique, it separated routing from the multicast protocol and can be easily adapted to support future work.

PIM-DM was changed to adapt it to a LEO satellite network. Changes to the specification's Join, State Refresh, Assert message types introduce greater network efficiency. Protocol changes facilitate rebuilding connections due to LEO satellite network dynamics and maintaining communications links with minimal packet losses while keeping the network state stable. Finally, the simple network protocol facilitates ground-to-satellite handoffs and has the losing satellite to forward packets to the gaining satellite.

5.3 Conclusions

A scalable, reliable protocol is a critical component in an information infrastructure. As implemented here, PIM-DM provides this capability in a LEO satellite network constellation.

System loading levels did not affect the protocol performance in any significant fashion. Instead, the protocol effectively adapted to increased loading levels and provided a reliable, sustainable transmission mechanism. Data-to-Overhead ratios were approximately 82%, Receive-to-Sent ratios were above 99.98%, and End-to-End delay times were approximately 76 ms.

Comparison of PIM-DM with both DVMRP and ODMRP shows that PIM-DM performed similarly or outperformed the other protocols across loading levels. PIM-DM's advantages stem from improvements in the protocol specification and the ability to separate the multicasting protocol from the routing protocol.

5.4 Future Research

Many facets of PIM-DM implementation lend themselves to areas for future research and improvement. This implementation of PIM-DM laid the groundwork for future work, but was not as robust as it could have been. The most obvious future research effort is to modify PIM-DM to implement PIM-SM. The dense nature of PIM-DM is not suitable for all tasks and having a model that works for both PIM-DM and PIM-SM would provide essential flexibility.

5.4.1 Routing

The most critical improvement would be to improve the unicast routing algorithm. Routing protocol limitations restricted the ability of PIM-DM to function within the confines of the 60-degree latitude restriction and resulted in possible one-way routes. A more complete routing algorithm should provide alternate routes between source/destination pairs and be able to intelligently route around or across the 60-degree boundary.

5.4.2 Satellite Constellations

Two areas should be explored for the satellite constellation: failures and alternate constellations. Adding satellite failures to the model would introduce real-world uncertainty and provide a way to explore various failure/overload scenarios.

Second, the LEO constellation modeled for this work was based on the Iridium® LEO constellation. Broadening the satellite constellation model to include additional LEO, MEO, and GEO systems, would allow data to be gathered across a wider range of applications and network configurations.

5.4.3 PIM-DM

The model made certain assumptions about the satellite-to-ground connection. Removing the restriction on the number of ground nodes per satellite would bring the PIM-DM implementation closer to the ODMRP and DVMRP implementations. This change would provide another method to increase the system load and introduce additional ground node placement topologies.

5.4.4 OPNET®

The main limiting factor in simulation capability is memory allocation. This problem is not initially obvious when the simulation size is small, but rapidly comes to the forefront once the simulations become more complex and incorporate more entities. The memory allocation problem appears to be linked to dynamic allocations. A potential fix would be the implementation of a static memory allocation library which provides the same capabilities as OPNET® memory functions, while satisfying requests from the static pool.

Appendix A. Data

Table A-1 PIM-DM, One-to-Many, Sparse Mode

Members	Loading	Data to Overhead		Receive to Sent		End to End Delay	
		μ	σ	μ	σ	μ	σ
5	1	0.645125	0.011225	0.999929	3.52E-05	0.074072	0.004229
	5	0.848971	0.004705	0.99993	3.48E-05	0.074087	0.004231
	20	0.901864	0.002488	0.99993	3.45E-05	0.074205	0.004231
10	1	0.666667	0.019998	0.999921	1.91E-05	0.076481	0.004438
	5	0.857332	0.006868	0.99992	1.96E-05	0.076488	0.004403
	20	0.905922	0.002653	0.999919	2.05E-05	0.076808	0.004629
15	1	0.679722	0.027768	0.999937	1.8E-05	0.080895	0.004922
	5	0.8602	0.008968	0.999937	1.84E-05	0.080937	0.004855
	20	0.90551	0.002491	0.999939	1.73E-05	0.081425	0.005086
20	1	0.673182	0.029927	0.999928	2.91E-05	0.079116	0.005224
	5	0.85886	0.009595	0.999926	2.99E-05	0.079284	0.005246
	20	0.905304	0.002864	0.999927	2.81E-05	0.07968	0.005472
30	1	0.679152	0.006834	0.999936	1.8E-05	0.072389	0.007077
	5	0.860104	0.002133	0.999938	1.83E-05	0.072567	0.007136
	20	0.905881	0.00096	0.999936	1.89E-05	0.073222	0.007177
40	1	0.695001	0.027007	0.999931	1.35E-05	0.075045	0.008912
	5	0.86491	0.008808	0.999932	1.48E-05	0.075258	0.008959
	20	0.907436	0.002184	0.999931	1.42E-05	0.076121	0.009006

Table A-2 PIM-DM, One-to-Many, Dense Mode

Members	Loading	Data to Overhead		Receive to Sent		End to End Delay	
		μ	σ	μ	σ	μ	σ
5	1	0.672882	0.021673	0.999917	1.47E-05	0.076457	0.011821
	5	0.858741	0.007152	0.999918	1.43E-05	0.076599	0.011862
	20	0.90546	0.002038	0.999917	1.3E-05	0.077207	0.01214
10	1	0.690245	0.013385	0.999904	1.84E-05	0.069865	0.007173
	5	0.864383	0.004054	0.999902	1.76E-05	0.06993	0.007151
	20	0.907045	0.001159	0.999903	1.68E-05	0.071026	0.007653
15	1	0.707637	0.027502	0.999896	1.55E-05	0.069522	0.006132
	5	0.86912	0.008527	0.999898	1.83E-05	0.069488	0.006277
	20	0.908503	0.002295	0.9999	1.69E-05	0.071082	0.006625
20	1	0.736398	0.016008	0.999878	1.69E-05	0.075779	0.006149
	5	0.877795	0.003929	0.999881	1.96E-05	0.076361	0.006381
	20	0.9104	0.001133	0.999884	1.6E-05	0.078663	0.007149
30	1	0.728484	0.029106	0.999892	9.82E-06	0.080094	0.003005
	5	0.875538	0.008611	0.999893	1.14E-05	0.07906	0.003245
	20	0.910136	0.002066	0.999886	1.36E-05	0.083389	0.003349
40	1	0.738177	0.04733	0.999842	4.01E-05	0.081038	0.001148
	5	0.877583	0.013682	0.999841	4.07E-05	0.080227	0.002104
	20	0.910808	0.003914	0.999847	5.25E-05	0.086862	0.003255

Table A-3 PIM-DM, Many-to-Many, All-to-All

Members	Loading	Data to Overhead		Receive to Sent		End to End Delay	
		μ	σ	μ	σ	μ	σ
5	1	0.686069	0.012623	0.999741	3.56E-05	0.074678	0.003759
	5	0.862722	0.003732	0.999743	3.3E-05	0.074924	0.00384
	20	0.906888	0.000947	0.99975	3.06E-05	0.075668	0.003781
10	1	0.721707	0.001299	0.999779	1.77E-05	0.073389	7.38E-05
	5	0.873721	0.000846	0.999781	2.03E-05	0.073807	0.000166
	20	0.909705	0.00057	0.999792	1.63E-05	0.076244	0.000186
15	1	0.715789	0.007028	0.999379	7.36E-05	0.077491	0.013126
	5	0.871726	0.001608	0.999388	5.16E-05	0.074182	0.002132
	20	0.909429	0.000147	0.999285	8.73E-05	0.086557	0.000941

Note: An * after the percentage denotes the effect was significant based on the computed F-Test

Table A-4 ANOVA Analysis for Sparse/Dense 5-10-15 Trials

		Data to Overhead	Receive to Sent	End to End Delay
Main Effects	Density	0.39%*	22.48%*	10.14%*
	Loading	97.12%*	0.01%	0.20%
	Membership	0.45%*	4.10%	1.60%
Second Order	Density-Loading	0.26%*	0.01%	0.06%
	Density-Membership	0.00%	6.01%	13.52%*
	Loading-Membership	0.30%	0.10%	0.03%
Third Order	Density-Loading-Membership	0.00%	0.03%	0.00%
	Unaccounted	1.49%	67.27%	74.45%

Table A-5 ANOVA Analysis for Sparse/Dense 20-30-40 Trials

		Data to Overhead	Receive to Sent	End to End Delay
Main Effects	Density	1.79%*	53.25%*	10.11%*
	Loading	93.31%*	0.01%	2.96%
	Membership	0.09%	6.97%*	1.95%
Second Order	Density-Loading	1.29%*	0.01%	1.50%
	Density-Membership	0.05%	5.84%*	12.42%*
	Loading-Membership	0.07%	0.12%	0.40%
Third Order	Density-Loading-Membership	0.03%	0.14%	0.26%
	Unaccounted	3.39%	33.65%	70.41%

Table A-6 ANOVA Analysis for All-All Trials

		Data to Overhead	Receive to Sent	End to End Delay
Main Effects	Density	0.79%	59.21%	29.27%
	Loading	97.82%	0.05%	6.19%
	Membership	0.48%	13.17%	10.07%
Second Order	Density-Loading	0.00%	0.00%	0.00%
	Density-Membership	0.00%	0.00%	0.00%
	Loading-Membership	0.53%	0.11%	5.51%
	Third Order/Unaccounted**	0.04%	26.85%	39.03%

** Due to incomplete runs for the All-15 experiment, the ANOVA was performed on the average values. This provides an approximation to the true ANOVA and also means that the third order runs and unexplained variance where not differentiable.

Appendix B. Summary Data for DVMRP/ODMRP/PIM-DM

Table B-1 DVMRP and ODMRP Summary Data

Protocol	Num Senders	Density	Members	Workload	DtO Mean	RtS Mean	EtE Mean
DVMRP	All	Sparse	40	50	0.5391593	0.9093618	0.0674788
				80	0.538903	0.9072533	0.0679448
				100	0.5392928	0.9077908	0.0683458
			60	50	0.597763	0.9160045	0.0676585
				80	0.5984913	0.9135575	0.0681035
				100	0.5977803	0.9121645	0.0685913
			80	50	0.636377	0.9124918	0.0684905
				80	0.6374565	0.9105208	0.069002
				100	0.6374123	0.9093828	0.0691308
		Dense	40	50	0.6489225	0.8545995	0.0680788
				80	0.6540888	0.8564715	0.0686338
				100	0.6482903	0.845103	0.0684473
			60	50	0.7231955	0.8910408	0.0693663
				80	0.7232088	0.8903035	0.0695918
				100	0.7254208	0.8907043	0.06995
			80	50	0.762932	0.8634078	0.077326
				80	0.7646408	0.862169	0.075342
				100	0.7663993	0.8627335	0.0752533
		Sparse	5	50	0.2918173	0.9355495	0.067715
				80	0.300428	0.9435325	0.068539
				100	0.310231	0.9353775	0.0688448
			10	50	0.3954158	0.8879425	0.0659055
				80	0.3986423	0.8909978	0.0665855
				100	0.4006298	0.8895633	0.066917
			15	50	0.4444475	0.8914975	0.0683463
				80	0.4449353	0.8870763	0.0687383
				100	0.4455248	0.8908253	0.0692798
	1		5	50	0.2556253	0.977848	0.062661
				80	0.269738	0.978692	0.0642655
				100	0.2908495	0.9843625	0.0658015
			10	50	0.3445853	0.9556073	0.058301
				80	0.3469518	0.956679	0.059611
				100	0.3537525	0.9562634	0.0610335
			15	50	0.3713905	0.9583515	0.06176
				80	0.37197	0.9579768	0.0633753
				100	0.3727598	0.9572353	0.0647345
ODMRP			5	50	0.2215334	0.8657636	0.060657
				80	0.2314064	0.902928	0.0660682
				100	0.2390862	0.9068568	0.0699302
			10	50	0.2390888	0.9233102	0.0578654
				80	0.2338903	0.9026015	0.0620688
				100	0.239543	0.9280018	0.0648145
			15	50	0.278142	0.9288543	0.0562365
				80	0.2508378	0.918599	0.064088
				100	0.2422953	0.9288008	0.0679703
	All		5	50	0.114274	0.9966373	0.0700668
				80	0.111433	0.99156	0.0720168
				100	0.1128605	0.9918308	0.0730975
			10	50	0.1190933	0.9964323	0.0616515
				80	0.115407	0.9950645	0.0637918
				100	0.1143108	0.9933875	0.0648708
			15	50	0.124201	0.9954035	0.0642838
				80	0.1225075	0.9936523	0.0661175
				100	0.1223993	0.9917935	0.0677313

Table B-2 PIM-DM Summary Data

Protocol	Num Senders	Density	Members	Workload	DtO Mean	RtS Mean	EtE Mean
PIM-DM	One	Sparse	5	1	0.645125	0.999929	0.074072
				5	0.848971	0.99993	0.074087
				20	0.901864	0.99993	0.074205
			10	1	0.666667	0.999921	0.076481
				5	0.857332	0.99992	0.076488
				20	0.905922	0.999919	0.076808
			15	1	0.679722	0.999937	0.080895
				5	0.8602	0.999937	0.080937
				20	0.90551	0.999939	0.081425
			20	1	0.673182	0.999928	0.079116
				5	0.85886	0.999926	0.079284
				20	0.905304	0.999927	0.07968
			30	1	0.679152	0.999936	0.072389
				5	0.860104	0.999938	0.072567
				20	0.905881	0.999936	0.073222
			40	1	0.695001	0.999931	0.075045
				5	0.86491	0.999932	0.075258
				20	0.907436	0.999931	0.076121
PIM-DM	One	Dense	5	1	0.672882	0.999917	0.076457
				5	0.858741	0.999918	0.076599
				20	0.90546	0.999917	0.077207
			10	1	0.690245	0.999904	0.069865
				5	0.864383	0.999902	0.06993
				20	0.907045	0.999903	0.071026
			15	1	0.707637	0.999896	0.069522
				5	0.86912	0.999898	0.069488
				20	0.908503	0.9999	0.071082
			20	1	0.736398	0.999878	0.075779
				5	0.877795	0.999881	0.076361
				20	0.9104	0.999884	0.078663
			30	1	0.728484	0.999892	0.080094
				5	0.875538	0.999893	0.07906
				20	0.910136	0.999886	0.083389
			40	1	0.738177	0.999842	0.081038
				5	0.877583	0.999841	0.080227
				20	0.910808	0.999847	0.086862
PIM-DM	ALL	ALL	5	1	0.686069	0.999741	0.074678
				5	0.862722	0.999743	0.074924
				20	0.906888	0.99975	0.075668
			10	1	0.721707	0.999779	0.073389
				5	0.873721	0.999781	0.073807
				20	0.909705	0.999792	0.076244
			15	1	0.715789	0.999379	0.077491
				5	0.871726	0.999388	0.074182
				20	0.909429	0.999285	0.086557

Table B-3 DtO Comparison

	DVMRP	ODMRP	PIM-DM
High	0.7664	0.3728	0.9108
Low	0.2918	0.1114	0.6451
Average	0.5619	0.2300	0.8229

Table B-4 RtS Comparison

	DVMRP	ODMRP	PIM-DM
High	0.9435	0.9966	0.9999
Low	0.8451	0.8658	0.9993
Average	0.8951	0.9568	0.9999

Table B-5 EtE Comparison

	DVMRP	ODMRP	PIM-DM
High	0.0773	0.0731	0.0869
Low	0.0659	0.0562	0.0695
Average	0.0692	0.0643	0.0764

Appendix C. Availability of OPNET® Models and Source Code

OPNET® Models and source code are not included as part of this document. Interested parties should direct their inquiries to:

Dr. Richard Raines

AFIT/ENG

2950 P. Street

Wright-Patterson AFB, OH 45433-7765

Bibliography

[ANS02] Adams A., J. Nicholas, W. Siadak, "Protocol Independent Multicast – Dense Mode (PIM-DM) : Protocol Specification (Revised)," Draft-IETF-PIM-DM-NEW-V2-01.txt (February 2002).

[ANV99] Allman, M., NASA Glenn/Sterling Software, V. Paxson, ACIRI/ICSI, W. Stevens. "TCP Congestion Control," Request for Comments: 2581, 1-14 (April 1999).

[Alm00] Almeroth, Kevin C. "The Evolution of Multicast: From the MBone to Interdomain Multicast to Internet2 Deployment," IEEE Network, 10-20 (January/February 2000).

[BLS00] Bae, Sang Ho, Sung-Ju Lee, William Su, and Mario Gerla. "The Design, Implementation, and Performance Evaluation of the On-Demand Multicast Routing Protocol in Multihop Wireless Networks," IEEE Network, 70-77 (January/February 2000).

[CCS01] Chotikapong, Yotsapak, Haitham Cruickshank, and Zhili Sun. "Evaluation of TCP and Internet Traffic via Low Earth Orbit Satellites," IEEE Personal Communications, 28-34 (June 2001).

[Dee89] Deering, Stephen, "Host Extensions for IP Multicasting," Request for Comments: 1112, 1-17 (August 1989).

[DeC90] Deering, Stephen E. and David Cheriton. "Multicast Routing in Datagram Internetworks and Extended LANs," ACM Transactions on Computer Systems, [Vol. 8, No. 2]:85-110 (May 1990).

[DEF96] Deering, Stephen, Deborah Estrin, Dino Farinacci, Van Jacobson, Ching-Gung Liu, and Liming Wei. "The PIM Architecture for Wide-Area Multicast Routing," IEEE/ACM Transactions on Networking, [Vol. 4, No. 2,]:153-162 (April 1996).

[DEF99] Deering, S., D. Estrin, D. Farinacci, V. Jacobson, A. Helmy, D. Meyer, L. Wei, "Protocol Independent Multicast Version 2 Dense Mode Specification," Draft-IETF-PIM-V2-DM-03.txt (June 1999).

[EAB01] Ekici, Eylem, Ian F. Akyildiz, and Michael D. Bender. "A Distributed Routing Algorithm for Datagram Traffic in LEO Satellite Networks," IEEE/ACM Transactions on Networking, [Vol. 9, No. 2]:137-147 (April 2001).

[EAB02] Ekici, Eylem, Ian F. Akyildiz, and Michael D. Bender. "A Multicast Routing Algorithm for LEO Satellite IP Networks," IEEE/ACM Transactions on Networking, [Vol. 10, No. 2]:183-192 (April 2002).

[Elb87] Elbert, Bruce. Introduction to Satellite Communication. Norwood, MA: Artech House, 1987, 22-25.

[EFH98] Estrin, D., D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, L. Wei, “Protocol Independent Multicast-Sparse Mode (PIM-SM) : Protocol Specification,” RFC 2362 (June 1998).

[Eva99] Evans, B. G. Satellite Communications Systems. London, UK: Institute of Electrical Engineers, 1999, 250-255

[FeX97] Fenner, W., Xerox PARC. “Internet Group Management Protocol, Version 2,” Request for Comments: 2236, 1-24 (November 1997).

[FHH02] Fenner, B., M. Handley, H. Holbrook, I. Kouvelas, “Protocol Independent Multicast – Sparse Mode (PIM-SM) : Protocol Specification (Revised),” Draft-IETF-PIM-SM-V2-NEW-04.txt (May 2002).

[FiD01] Filali, Fethi, Walid Dabbous. “Issues on the IP Multicast Service Behaviour over the Next-Generation Satellite Terrestrial Hybrid Networks,” IEEE, 417-424 (2001).

[Fos01] Fossa, C. E. Jr., “A performance analysis of the IRIDIUM low earth satellite system,” Master’s thesis, Air Force Institute of Technology, 1998, AFIT/GE/ENG/98J-01.

[GhD99] Ghani, Nasir and Sudhir Dixit. “TCP/IP Enhancements for Satellite Networks,” IEEE Communications Magazine, 64-72 (July 1999).

[Jai91] Jain, Raj. The Art of Computer Systems Performance Analysis. New York: John Wiley & Sons, 1991, 348-366,420-421.

[KoF01] Koyabe, Martin W. and Godred Fairhurst. “Reliable multicast via satellite: a comparison survey and taxonomy,” International Journal of Satellite Communications, 19:3-28 (2001).

[McC00] McCreary, S., “Packet Length Distributions,” http://www.caida.org/analysis/AIX/plen_hist/index.xml (March 2000).

[Met99] Metz, Christopher. “TCP Over Satellite…The Final Frontier,” IEEE Internet Computing, 76-80 (January/February 1999).

[Met02] Metz, Christopher. “IP Anycast – Point-to-(Any) Point Communication,” IEEE Internet Computing, 94-98 (March/April 2002).

[PKK01] Park, Juyoung, Seok Joo Koh, Shin Gak Kang, and Dae Young Kim. "Multicast Delivery Based on Unicast and Subnet Multicast," IEEE Communications Letters, [Vol 5, No 4]:181-183 (April 2001).

[PeD00] Peterson, L., and B. Davis. Computer Networks. San Diego:Morgan Kaufmann Publishers, 19 (2000).

[Pra99] Pratt, S. R., "A performance analysis of dynamic routing algorithms in an Iridium-like low earth orbit satellite system," Master's thesis, Air Force Institute of Technology, 1999. AFIT/GCE/ENG/99M-04.

[QCA01] Quinn, B., Celox Networks, K. Almeroth, UC-Santa Barbara, "IP Multicast Applications: Challenges and Solutions," Request for Comments: 3170, 1-28 (September 2001).

[RaS96] Ramanathan, S., Martha Steenstrup. "A survey of routing techniques for mobile communications networks," Mobile Networks and Applications, 1: 89-104 (1996).

[Rod89] Rody, Dennis. Satellite Communications. New York: McGraw-Hill, 1989, 67-88.

[SaM00] Sahasrabuddhe, Laxman H. and Biswanath Mukherjee. "Multicast Routing Algorithms and Protocols: A Tutorial," IEEE Network, 90-102 (January/February 2000).

[Ste94] Stevens, W. Richard. TCP/IP Illustrated Volume 1 – The Protocols. Reading, Mass: Addison-Wesley, 1994. 171, 169

[Tay98] Taylor, Ed. TCP/IP Complete. New York: McGraw-Hill, 1998. 38,205,208,210

[TMH00] Thaler, D., Microsoft, C. Hopps, NextHop Technologies. "Multipath Issues in Unicast and Multicast Next-Hop Selection," Request for Comments: 2991, 1-9 (November 2000).

[Tho01] Thomas, R. W., "Multicasting algorithms for mobile satellite communication networks," Master's thesis, Air Force Institute of Technology, 2001, AFIT/GCE/ENG/01M-04.

[Tom98] Tomasi, Wayne. Advanced Electronic Communications Systems, Upper Saddle River, NJ: Prentice Hall, 1998, 302-306.

[WaC01] Wang, Yu, Weidong Chen. "Supporting IP Multicast for Mobile Hosts," Mobile Networks and Applications, 6:57-66 (2001).

[WCA01] Wood, L., A. Clerget, I. Andrikopoulos, G. Pavlou, and W. Dabbous. "IP routing issues in satellite constellation networks," International Journal of Satellite Communications, 19:69-92 (2001).

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 074-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to an penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
25-03-2003	**Master's Thesis**	Jun 2002 – Mar 2003

4. TITLE AND SUBTITLE	**5a. CONTRACT NUMBER**
PERFORMANCE ANALYSIS OF PROTOCOL INDEPENDENT MULTICASTING-DENSE MODE IN LOW EARTH ORBIT SATELLITE NETWORKS	**5b. GRANT NUMBER**
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	**5d. PROJECT NUMBER**
Saeger, Mark D., 1Lt, USAF	**5e. TASK NUMBER**
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)	**8. PERFORMING ORGANIZATION REPORT NUMBER**
Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way, Building 640 WPAFB OH 45433-7765	AFIT/GCE/ENG/03-03
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	**10. SPONSOR/MONITOR'S ACRONYM(S)**
JWAC/J53 Attn: Maj. James E. Kent 18385 Frontage Road Comm: 540-255-4120 Dahlgren VA 22448-5500 e-mail: jkent@jwac.osis.gov	**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

12. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This research explored the implementation of Protocol Independent Multicasting – Dense Mode (PIM-DM) in a LEO satellite constellation. PIM-DM is a terrestrial protocol for distributing traffic efficiently between subscriber nodes by combining data streams into a tree-based structure, spreading from the root of the tree to the branches. Using this structure, a minimum number of connections are required to transfer data, decreasing the load on intermediate satellite routers.

The PIM-DM protocol was developed for terrestrial systems and this research implemented an adaptation of this protocol in a satellite system. This research examined the PIM-DM performance characteristics which were compared to earlier work for On-Demand Multicast Routing Protocol (ODMRP) and Distance Vector Multicasting Routing Protocol (DVMRP) – all in a LEO satellite network environment.

Experimental results show that PIM-DM is extremely scalable and has equivalent performance across diverse workloads. Three performance metrics are used to determine protocol performance in the dynamic LEO satellite environment, including Data-to-Overhead ratio, Received-to-Sent ratio, and End-to-End Delay. The OPNET® simulations show that the PIM-DM Data-to-Overhead ratio is approximately 80% and the protocol reliability is extremely high, achieving a Receive-to-Sent ratio of 99.98% across all loading levels. Finally, the PIM-DM protocol introduces minimal delay, exhibiting an average End-to-End Delay of approximately 76 ms; this is well within the time necessary to support real-time communications. Though fundamental differences between the DVMRP, ODMRP, and PIM-DM implementations precluded a direct comparison for each experiment, by comparing average values, PIM-DM generally provides equivalent or better performance.

15. SUBJECT TERMS
Multicast, Protocol Independent Multicast – Dense Mode (PIM-DM)

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Richard A Raines, AD-23, USAF (ENG)
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(Include area code)*
U	U	U	UU	130	(937) 255-6565, ext 4278; e-mail: Richard.Raines@afit.edu

Student	Faculty Advisor	Thesis Designator		Keyword #1	Keyword #2
M.D. Saeger	Raines, R.A.	AFIT/GCE/ENG/03-03		Multicast	PIM-DM
Sponsor (MAJCOM)	**Agent**	**Phone**	**Program**		**Funding**
JWAC/J53	Kent, J. E.	(540)-255-4120			$0

Title: Performance Analysis of Protocol Independent Multicasting-Dense Mode in Low Earth Orbit Satellite Networks

Subject: Examine the behavior of the PIM-DM multicasting protocol in a LEO satellite environment.

Air Force Program Description:

Communication networks are increasingly global, leading to the logical progression of using space-based assets to facilitate the communication infrastructure. The medium of space provides capabilities that are not possible for terrestrial systems - most importantly global coverage immune to the vagaries of border demarcations by sovereign states. Global reach communication does come at a price - space-based assets are extremely expensive as compared to their terrestrial counterparts. As such, the most efficient use of the space-based asset is paramount. This thesis explored the implementation of Protocol Independent Multicasting - Dense Mode (PIM-DM) in a LEO satellite constellation. PIM-DM is a protocol definition for distributing traffic efficiently between subscriber nodes by combining data streams into a tree-based structure, spreading from the root of the tree to the branches. The efficiency introduced by the tree is that the minimum number of connections are necessary to transfer the data, thereby decreasing the load on intermediate satellite routers.

Impact Statement:
This research explored the implementation of Protocol Independent Multicasting – Dense Mode (PIM-DM) in a LEO satellite constellation. PIM-DM is a terrestrial protocol for distributing traffic efficiently between subscriber nodes by combining data streams into a tree-based structure, spreading from the root of the tree to the branches. The PIM-DM protocol was developed for terrestrial systems and this research implemented an adaptation of this protocol in a satellite system. This research examined the PIM-DM performance characteristics which were compared to earlier work for On-Demand Multicast Routing Protocol (ODMRP) and Distance Vector Multicasting Routing Protocol (DVMRP) – all in a LEO satellite network environment.

Technical Abstract:
This research explored the implementation of Protocol Independent Multicasting – Dense Mode (PIM-DM) in a LEO satellite constellation. PIM-DM is a terrestrial protocol for distributing traffic efficiently between subscriber nodes by combining data streams into a tree-based structure, spreading from the root of the tree to the branches. Using this structure, a minimum number of connections are required to transfer data, decreasing the load on intermediate satellite routers.

The PIM-DM protocol was developed for terrestrial systems and this research implemented an adaptation of this protocol in a satellite system. This research examined the PIM-DM performance characteristics which were compared to earlier work for On-Demand Multicast Routing Protocol (ODMRP) and Distance Vector Multicasting Routing Protocol (DVMRP) – all in a LEO satellite network environment.

Experimental results show that PIM-DM is extremely scalable and has equivalent performance across diverse workloads. Three performance metrics are used to determine protocol performance in the dynamic LEO satellite environment, including Data-to-Overhead ratio, Received-to-Sent ratio, and End-to-End Delay. The OPNET® simulations show that the PIM-DM Data-to-Overhead ratio is approximately 80% and the protocol reliability is extremely high, achieving a Receive-to-Sent ratio of 99.98% across all loading levels. Finally, the PIM-DM protocol introduces minimal delay, exhibiting an average End-to-End Delay of approximately 76 ms; this is well within the time necessary to support real-time communications. Though fundamental differences between the DVMRP, ODMRP, and PIM-DM implementations precluded a direct comparison for each experiment, by comparing average values, PIM-DM generally provides equivalent or better performance.

Subject Terms: Multicast, Protocol Independent Multicasting – Dense Mode (PIM-DM)

Publications: None